

**PENGEMBANGAN SISTEM INFORMASI *CUSTOMER -  
TOUCHING APPLICATIONS* UNIT SIMPAN PINJAM  
MENGUNAKAN METODE RUP  
(STUDI PADA KPRI “GURU” KECAMATAN  
SUMBERMANJING WETAN)**

**SKRIPSI**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:  
Fatih Ulil Albab  
NIM: 145150400111071



SISTEM INFORMASI  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2018

**PENGESAHAN**

PENGEMBANGAN SISTEM *INFORMASI CUSTOMER - TOUCHING APPLICATIONS*  
UNIT SIMPAN PINJAM MENGGUNAKAN METODE RUP (STUDI PADA KPRI  
"GURU" KECAMATAN SUMBERMANJING WETAN)

**SKRIPSI**

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh :  
Fatih Ulii Albab  
NIM: 145150400111071

Skripsi ini telah diuji dan dinyatakan lulus pada  
29 Juni 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Yusi Tyroni Mursitvo, S.Kom., M.AB.

NIP: 198002282006041001

Djoko Pramono, S.T., M.Kom.

NIP: 197801082005011002

Mengetahui

Ketua Jurusan Sistem Informasi



Dr. Eng. Herman Tolle, S.T., M.T.

NIP: 19740823 200012 1 001

## IDENTITAS PENGUJI

Informasi pelaksanaan ujian dan penguji.

### JADWAL

Jadwal pelaksanaan ujian

Hari/Tanggal	JUMAT / 29 JUNI 2018
Tempat	FILKOM UB / F3.4
Waktu	07:30 - 08:45

### PENGUJI

Majelis penguji ujian skripsi



Issa Arwani, S.Kom, M.Sc  
(ke I) \* ketua majelis  
NIP. 19830922 201212 1 003



Agi Putra Kharisma, S.T, M.T  
(ke II)  
NIK. 201304 860430 1 001

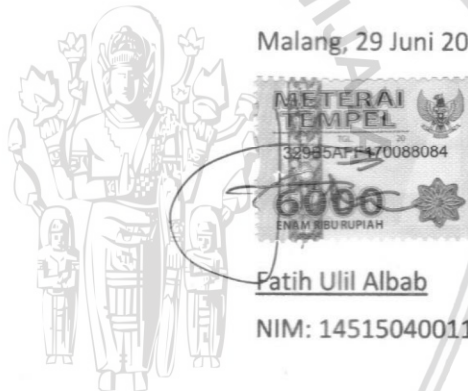


## PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 29 Juni 2018



Patih Ulil Albab

NIM: 145150400111071

## DAFTAR RIWAYAT HIDUP

### Data Pribadi

Nama lengkap : Fatih Ulil Albab  
 Tempat, tanggal lahir : Malan, 26 Juni 1995  
 Jenis Kelamin : Laki – laki  
 Status : Belum Kawin  
 Kewarganegaraan : Indonesia  
 Alamat : Jln. Darmasastra no 3 RT. 04 / RW.01  
 Sumbermanjing Wetan, Malang  
 No. Hp : 082257730729  
 Email : fatihulil@gmail.com



### Latar Belakang Pendidikan

#### Pendidikan Formal

2014 - 2018 : S1 Sistem Informasi Fakultas Ilmu Komputer  
 Universitas Brawijaya, IPK = 3,71 (Skala 4)  
 2011 - 2014 : SMA Negeri 1 Kepanjen  
 2008 – 2011 : SMP Negeri 1 Turen  
 2002 – 2008 : SD Negeri 1 Sumbermanjing Wetan  
 2000 – 2002 : TK Dharma Wanita Sumbermanjing Wetan

### Pengalaman Organisasi

1. Wakil Koordinator *Creative Design* “The Night of Rebirth” FILKOM 2016
2. Wakil Koordinator *Creative Design* “Jalan Sehat Dies Natalis” FILKOM 2016
3. Wakil Koordinator *Creative Design* “Filkom All Star” FILKOM 2016
4. Koordinator Desain dan Multimedia SIGATH 2015
5. Wakil Desain dan Multimedia WOW (Week of Welcoming) Sistem Informasi 2015
6. Anggota Divisi Humas Olimpiade PTIHK 2015
7. Ketua Sie Peribadatan Badan Dakwah Islam SMAN 1 Kepanjen 2013

## UCAPAN TERIMA KASIH

Penelitian ini tidak dapat diselesaikan tanpa dukungan dan bantuan dari orang-orang terdekat, sehingga saya ingin menyampaikan rasa terima kasih kepada:

1. Ibunda Umi Hani'in dan ayahanda Arifin Ihsan serta keluarga besar yang senantiasa memberikan semangat, do'a dan motivasi dalam menyelesaikan semua tugas dan tanggung jawab yang dimiliki, serta senantiasa memberikan contoh untuk menjadi pribadi yang disiplin dan bertanggung jawab.
2. Bapak Yusi Tyroni Mursityo, S.Kom., M.AB, selaku pembimbing satu yang telah mengajarkan ketegasan dan kedisiplinan dalam menyelesaikan kewajiban. Memberikan konsep pemikiran yang luas dalam mencari solusi dari sebuah permasalahan.
3. Bapak Djoko Pramono, S.T., M.Kom., selaku pembimbing dua yang telah mengajarkan ketelitian serta banyak memberikan ilmu, saran, dan motivasi untuk penyelesaian skripsi ini.
4. Segenap pengawas, pengurus harian dan petugas KPRI "GURU" Kecamatan Sumbermanjing Wetan yang telah mengizinkan serta membantu penulis untuk melakukan penelitian ini.
5. Sahabat – sahabat terdekat Teta Fatma, Ahmad Zaky, Edo Pratma, Moch. Ali Hafizd, Nanda Adhi, Farhani Ayu, Ardyani Nalendri, Dhena Kamalia, Sarah Rahmi, Fathania Fadilla, Sanya Nabila, Arya Henda dan teman – teman lainnya yang tidak dapat saya sebutkan satu per satu. Terima kasih atas pengalaman berharga dan bantuan yang telah diberikan kepada penulis semasa perkuliahan.
6. Teman-teman Jurusan Sistem Informasi, Fakultas Ilmu Komputer, yang tidak dapat saya sebutkan namanya satu per satu, yang telah memberikan saya dukungan moril, menjadi teman dan sahabat yang baik dan menyenangkan selama masa perkuliahan.

Semoga Allah SWT melimpahkan rahmat berlipat ganda kepada seluruh pihak yang membantu dan berkontribusi pada proses penelitian ini. Tidak lupa saya sampaikan bahwa penelitian yang telah dilakukan memiliki banyak kekurangan, diskusi mengenai penelitian lanjutan, saran, dan kritik yang membangun merupakan beberapa hal yang saya harapkan dapat disampaikan kepada saya.



## ABSTRAK

**Fatih Ulil Albab, Pengembangan Sistem Informasi Customer-Touching Applications Unit Simpan Pinjam Menggunakan Metode Rup (Studi Pada Kpri “Guru” Kecamatan Sumbermanjing Wetan)”**

**Dosen Pembimbing: Yusi Tyroni Mursityo, S.Kom.,M.AB dan Djoko Pramono, S.T., M.Kom.**

KPRI “GURU” Kecamatan Sumbermanjing Wetan adalah sebuah koperasi yang beranggotakan guru pegawai negeri di Kecamatan Sumbermanjing Wetan. Salah satu bagian dari koperasi ini adalah unit simpan pinjam merupakan pelayanan jasa simpan dan pinjam bagi anggotanya. Saat ini proses bisnis unit simpan pinjam masih manual, dimana seluruh aktivitas hanya dapat dilakukan di kantor KPRI. Hal ini menimbulkan permasalahan, karena kondisi alam kecamatan Sumbermanjing Wetan yang luas dan berbukit-bukit memungkinkan anggota KPRI untuk berpindah ke koperasi desa yang lebih mudah diakses. Oleh karena itu sebuah sistem informasi yang berdasar pada strategi *Customer Relationship Management (CRM)* khususnya *Service Automation* dibutuhkan untuk mempertahankan loyalitas dari anggota koperasi. Luaran dari penelitian ini sebuah sistem informasi *Customer - Touching Applications* yang dapat memudahkan anggota untuk beraktivitas didalam unit simpan pinjam. Dalam pengembangan sistem tersebut menggunakan metode RUP karena dapat mengakomodasi perubahan selama pengembangan, karena sistem yang dikembangkan termasuk dalam sistem keuangan yang fungsionalitasnya sangat diperhatikan. Metode *Rational Unified Proses (RUP)* yang dilakukan adalah fase insepisi, elaborasi, konstruksi dan transisi. Sistem yang dikembangkan memiliki 23 kebutuhan fungsional untuk mendukung proses bisnis pada unit simpan pinjam. Pada pengujian sistem dilakukan dengan uji *black box* terdapat 67 *test case* untuk menguji kebutuhan fungsional sistem dengan hasil 100% dan *compatibility* untuk menguji kebutuhan non fungsional dengan hasil 2 *critical issue*. Pengujian UAT digunakan untuk mengetahui apakah sistem diterima oleh pengguna akhir dan menghasilkan nilai penerimaan sebesar 83,123%.

Kata kunci: *Customer Relationship Management (CRM), Rational Unified Process (RUP), sistem informasi, koperasi, simpan pinjam*

## ABSTRACT

**Fatih Ulil Albab, Pengembangan Sistem Informasi Customer-Touching Applications Unit Simpan Pinjam Menggunakan Metode Rup (Studi Pada Kpri "Guru" Kecamatan Sumbermanjing Wetan)"**

**Supervisors: Yusi Tyroni Mursityo, S.Kom.,M.AB dan Djoko Pramono, S.T., M.Kom.**

*KPRI "GURU" Kecamatan. Sumbermanjing Wetan is a corporation consisting of civil servant teachers in Sumbermanjing Wetan. One part of this corporation is a unit simpan pinjam, a service of savings and loans for its members. The currently savings and loan unit business process is still manually, where all activities can only be done at KPRI office. This raises the problem, because the natural condition of Sumbermanjing Wetan sub-district which is vast and hilly area allows KPRI members to move to more accessible corporation. Therefore an information system based on Customer Relationship Management (CRM) strategy especially Service Automation is needed to maintain the loyalty of corporation members. Result from this study is a Customer - Touching Applications system to ease members in the savings and loan activity. In the developing system RUP method used because it can accommodate changes during development when developed the system included in the financial system which the functionality is a great concerned. The Rational Unified Process (RUP) method is an inception, elaboration, construction and transition. The developed system has 23 functional requirements to support business processes on savings and credit units. Testing on prototype system were performed by black box test there are 67 test case to test functional requirement system with 100% result and compatibility to test non functional requirement with result of 2 critical issue. UAT testing is used to determine whether the system is accepted by the end user and generates an acceptance value of 83.123%*

**Keywords:** *Customer Relationship Management (CRM), Metode Rational Unified Proses (RUP), information systems, cooperatives, savings and loans*



## KATA PENGANTAR

Segala puji bagi Allah SWT atas limpahan rahmat yang diberikan sehingga penulis dapat menyelesaikan skripsi berjudul *“Pengembangan Sistem Informasi Customer-Touching Applications Unit Simpan Pinjam Menggunakan Metode Rup (Studi Pada Kpri “Guru” Kecamatan Sumbermanjing Wetan)”*.

Dalam proses pengerjaan penelitian ini, peneliti banyak mendapatkan ilmu tidak diajarkan pada materi perkuliahan. Selain itu peneliti juga mendapatkan pengalaman untuk berinteraksi dan berkomunikasi dengan banyak orang, menyelaraskan berbagai persepsi, menerima kritik dan saran yang ada serta menanamkan rasa tanggung jawab pada diri sendiri. Penelitian ini tidak dapat diselesaikan tanpa dukungan dan bantuan dari orang-orang terdekat, sehingga saya ingin menyampaikan rasa terima kasih kepada:

7. Bapak Yusi Tyroni Mursityo, S.Kom., M.AB, selaku pembimbing satu yang telah mengajarkan ketegasan dan kedisiplinan dalam menyelesaikan kewajiban. Memberikan konsep pemikiran yang luas dalam mencari solusi dari sebuah permasalahan.
8. Bapak Djoko Pramono, S.T., M.Kom., selaku pembimbing dua yang telah mengajarkan ketelitian serta banyak memberikan ilmu, saran, dan motivasi untuk penyelesaian skripsi ini.
9. Bapak Satrio Agung W., S.Kom, M.Kom. selaku dosen penasehat akademik yang senantiasa memberikan masukan dan solusi permasalahan akademik.
10. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya Malang.
11. Bapak Herman Tolle, Dr. Eng., S.T, M.T selaku Ketua Jurusan Sistem Informasi Universitas Brawijaya Malang.
12. Bapak Suprpto, S.T, M.T selaku Ketua Program Studi Sistem Informasi Universitas Brawijaya Malang.
13. Ibunda Umi Hani'in dan ayahanda Arifin Ihsan serta keluarga besar yang senantiasa memberikan semangat, do'a dan motivasi dalam menyelesaikan semua tugas dan tanggung jawab yang dimiliki, serta senantiasa memberikan contoh untuk menjadi pribadi yang disiplin dan bertanggung jawab.
14. Segenap pengawas, pengurus harian dan petugas KPRI “GURU” Kecamatan Sumbermanjing Wetan yang telah mengizinkan serta membantu penulis untuk melakukan penelitian ini.
15. Sahabat – sahabat terdekat Teta Fatma, Ahmad Zaky, Edo Pratma, Moch. Ali Hafizd, Nanda Adhi, Farhani Ayu, Ardyani Nalendri, Dhena Kamalia, Sarah Rahmi, Fathania Fadilla, Sanya Nabila, Arya Henda dan teman – teman lainnya yang tidak dapat saya sebutkan satu per satu. Terima kasih atas pengalaman berharga dan bantuan yang telah diberikan kepada penulis semasa perkuliahan.

16. Teman-teman Jurusan Sistem Informasi, Fakultas Ilmu Komputer, yang tidak dapat saya sebutkan namanya satu per satu, yang telah memberikan saya dukungan moril, menjadi teman dan sahabat yang baik dan menyenangkan selama masa perkuliahan.

Semoga Allah SWT melimpahkan rahmat berlipat ganda kepada seluruh pihak yang membantu dan berkontribusi pada proses penelitian ini. Tidak lupa saya sampaikan bahwa penelitian yang telah dilakukan memiliki banyak kekurangan, diskusi mengenai penelitian lanjutan, saran, dan kritik yang membangun merupakan beberapa hal yang saya harapkan dapat disampaikan kepada saya.

Malang, 29 Juni 2018

Penulis  
fatihulil@gmail.com



## DAFTAR ISI

PERNYATAAN ORISINALITAS .....	iv
KATA PENGANTAR.....	vi
ABSTRAK .....	vii
ABSTRACT .....	viii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xvi
DAFTAR GAMBAR .....	xix
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	3
1.3 Tujuan .....	3
1.4 Manfaat.....	4
1.5 Batasan masalah .....	4
1.6 Sistematika pembahasan.....	4
BAB 2 LANDASAN KEPUSTAKAAN .....	6
2.1 Kajian Pustaka .....	6
2.2 Koperasi .....	10
2.3 KPRI “GURU” Kecamatan Sumbermanjing Wetan.....	10
2.4 Simpan Pinjam .....	11
2.5 <i>Customer Relationship Management (CRM)</i> .....	11
2.5.1 <i>Fase Customer Relationship Management (CRM)</i> .....	12
2.5.2 <i>Jenis Customer Relationship Management (CRM)</i> .....	13
2.6 Operational CRM .....	14
2.6.1 <i>Service Automation (SA)</i> .....	15
2.6.2 <i>Customer-Touching Applications</i> .....	15
2.7 Proses Bisnis.....	16
2.7.1 <i>Permodelan proses bisnis as-is</i> .....	16
2.7.2 <i>Permodelan proses bisnis to-be</i> .....	16
2.8 <i>Business Process Modelling Notation (BPMN)</i> .....	17
2.8.1 <i>Flow Objects</i> .....	17
2.8.2 <i>Connection</i> .....	19

2.8.3 Swimlanes .....	20
2.8.4 Artifact .....	20
2.9 Permodelan <i>Use Case</i> .....	21
2.9.1 Identifikasi Tipe Pemangku Kepentingan .....	21
2.9.2 Analisis Masalah .....	22
2.9.3 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna .....	22
2.9.4 Identifikasi Fitur .....	23
2.9.5 Identifikasi Kebutuhan Fungsional dan Kebutuhan Non Fungsional .....	24
2.10 Unified Modeling Language (UML) .....	24
2.10.1 Use Case Diagram .....	24
2.10.2 Activity Diagram .....	26
2.10.3 Sequence Diagram .....	27
2.10.4 Class Diagram .....	28
2.11 Physical Data Model .....	28
2.12 Model-View-Controller .....	29
2.13 Rational Unified Process (RUP) .....	30
2.13.1 Fase – Fase pada Metode RUP .....	31
2.13.2 Aktivitas dalam Metode RUP .....	33
2.14 Pengujian .....	34
2.14.1 Black Box Testing .....	34
2.14.2 Compatibility Testing .....	34
2.14.3 User Acceptance Testing .....	35
BAB 3 METODOLOGI .....	37
3.1 Tempat dan Waktu Penelitian .....	37
3.2 Langkah – Langkah Penelitian .....	37
3.3 Studi Literatur .....	38
3.4 Pengumpulan Data .....	38
3.4.1 Wawancara .....	38
3.4.2 Observasi .....	38
3.5 Fase Insepsi .....	38
3.5.1 Permodelan Proses Bisnis .....	39
3.5.2 Analisi Kebutuhan .....	39

3.6 Fase Elaborasi .....	39
3.6.1 Perancangan Sistem .....	39
3.7 Fase Konstruksi .....	39
3.7.1 Implementasi .....	40
3.7.2 Pengujian Sistem .....	40
3.8 Fase Transisi .....	40
3.8.1 Pengujian UAT .....	40
3.9 Kesimpulan dan Saran .....	40
BAB 4 ANALISIS .....	41
4.1 Analisis Proses Bisnis .....	41
4.1.1 Pemodelan Proses Bisnis <i>As-Is</i> .....	41
4.1.2 Analisis Permasalahan .....	46
4.1.3 Permodelan Proses Bisnis <i>To-be</i> .....	47
4.1.4 Analisa Perbaikan Proses bisnis .....	53
4.2 Analisis Kebutuhan .....	54
4.2.1 Identifikasi Pemangku Kepentingan .....	54
4.2.2 Kebutuhan Pemangku Kepentingan .....	55
4.2.3 Identifikasi Pengguna .....	56
4.2.4 Identifikasi Fitur .....	56
4.2.5 Spesifikasi Kebutuhan .....	57
4.3 Permodelan Use Case Diagram .....	62
4.3.1 Skenario <i>Use Case</i> .....	63
4.3.2 Permodelan <i>Activity Diagram</i> .....	78
BAB 5 PERANCANGAN .....	99
5.1 <i>Sequence Diagram</i> .....	99
5.1.1 <i>Sequence Diagram</i> Melihat Informasi Simpanan .....	99
5.1.2 <i>Sequence Diagram</i> Melihat Detail Simpanan .....	100
5.1.3 <i>Sequence Diagram</i> Melihat Informasi Pinjaman .....	101
5.1.4 <i>Sequence Diagram</i> Melihat Detail Pinjaman .....	101
5.1.5 <i>Sequence Diagram</i> Mengajukan Pinjaman .....	102
5.1.6 <i>Sequence Diagram</i> Menghitung Estimasi .....	103
5.1.7 <i>Sequence Diagram</i> Mengubah <i>Password</i> .....	104

5.1.8 <i>Sequence Diagram</i> Mengunduh Form Pinjaman .....	105
5.1.9 <i>Sequence Diagram</i> Melihat Detail Pengajuan .....	105
5.2 <i>Design Class Diagram</i> .....	106
5.3 Perancangan Basis Data .....	107
5.4 Perancangan Algoritma .....	108
5.5 Perancangan Antarmuka .....	110
BAB 6 IMPLEMENTASI .....	116
6.1 Spesifikasi Sistem .....	116
6.1.1 Spesifikasi Perangkat Keras .....	116
6.1.2 Spesifikasi Perangkat Lunak .....	116
6.2 Implementasi <i>Class</i> .....	117
6.3 Implementasi Basis Data .....	118
6.4 Implementasi Algoritma .....	118
6.4.1 Algoritma Melihat Informasi Simpanan .....	118
6.4.2 Algoritma Melihat Detail Simpanan .....	119
6.4.3 Algoritma Mengajukan pinjaman .....	120
6.4.4 Algoritma Melihat Detail Pengajuan .....	120
6.5 Implementasi Antarmuka .....	120
6.5.1 Implementasi Antarmuka <i>Login</i> Anggota .....	121
6.5.2 Implementasi Antarmuka <i>Home</i> Anggota .....	121
6.5.3 Implementasi Antarmuka Simpanan Anggota .....	122
6.5.4 Implementasi Antarmuka Detail Simpanan .....	122
6.5.5 Implementasi Antarmuka Pinjaman .....	123
6.5.6 Implementasi Antarmuka Detail Pinjaman .....	124
6.5.7 Implementasi Antarmuka Pengajuan Pinjaman .....	124
6.5.8 Implementasi Antarmuka Form Pengajuan .....	125
6.5.9 Implementasi Antarmuka Edit <i>Password</i> .....	126
6.5.10 Implementasi Antarmuka Detail Pengajuan .....	126
BAB 7 PENGUJIAN .....	127
7.1 Pengujian <i>Black-Box</i> .....	127
7.1.1 Perancangan Pengujian <i>Black-Box</i> .....	127
7.1.2 Hasil Pengujian <i>Black-box</i> .....	135



7.2 Pengujian <i>Compatibility</i> .....	140
7.3 Pengujian UAT.....	141
BAB 8 KESIMPULAN DAN SARAN .....	145
8.1 Kesimpulan.....	145
8.2 Saran .....	146
DAFTAR PUSTAKA.....	147
LAMPIRAN WAWANCARA .....	149
LAMPIRAN PENGUJIAN UAT.....	152



## DAFTAR TABEL

Table 2.1 Kajian Pustaka .....	8
Table 2.2 Tipe <i>Event</i> dalam BPMN .....	17
Table 2.3 Tipe <i>Activity</i> dalam BPMN .....	18
Table 2.4 Tipe <i>Gateway</i> dalam BPMN .....	19
Table 2.5 Tipe <i>Connection</i> dalam BPMN .....	19
Table 2.6 Tipe <i>Swimlanes</i> dalam BPMN .....	20
Table 2.7 Tipe <i>Artifact</i> dalam BPMN .....	20
Table 2.8 Analisis Tipe Pemangku Kepentingan .....	22
Table 2.9 Analisis Masalah .....	22
Table 2.10 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna .....	23
Table 2.11 Fitur Sistem .....	23
Table 2.12 Simbol <i>Use Case Diagram</i> .....	24
Table 2.13 Skenario <i>Use Case</i> .....	25
Table 2.14 Simbol <i>Activity Diagram</i> .....	26
Table 2.15 Simbol dalam <i>Sequence Diagram</i> .....	27
Table 2.16 Simbol <i>Class Diagram</i> .....	28
Table 2.17 Kriteria <i>User Acceptance Testing</i> .....	35
Table 4.1 Analisis Masalah Pengajuan Pinjaman .....	46
Table 4.2 Analisis permasalahan dan waktu yang dibutuhkan .....	47
Table 4.3 Perbandingan Proses Bisnis .....	53
Table 4.4 Tabel Kebutuhan Pemangku Kepentingan .....	54
Table 4.5 Table Kebutuhan Pemangku Kepentingan .....	55
Table 4.6 Tabel Peran Pengguna .....	56
Table 4.7 Tabel Fitur Pengguna .....	57
Table 4.8 Kebutuhan Fungsional User .....	58
Table 4.9 Kebutuhan Fungsional Anggota (Fase Insepsi) .....	58
Table 4.10 Tambahan Kebutuhan Fungsional Anggota (Fase Konstruksi) .....	59
Table 4.11 Kebutuhan Fungsional Petugas .....	60
Table 4.12 Kebutuhan Non Fungsional .....	61
Table 4.13 Skenario <i>Use Case</i> Login .....	64
Table 4.14 Skenario <i>Use Case</i> melihat informasi simpanan .....	64

Table 4.15 Skenario <i>Use Case</i> melihat pinjaman .....	65
Table 4.16 Skenario <i>Use Case</i> mengajukan pinjaman (Fase Insepsi).....	65
Table 4.17 Skenario <i>Use Case</i> mengajukan pinjaman (Fase Elaborasi) .....	66
Table 4.18 Skenario <i>Use Case</i> konfirmasi pinjaman .....	67
Table 4.19 Skenario <i>Use Case</i> melihat data simpanan .....	67
Table 4.20 Skenario <i>Use Case</i> Merubah Data Simpan .....	68
Table 4.21 Skenario <i>Use Case</i> Melihat Data Pinjaman .....	69
Table 4.22 Skenario <i>Use Case</i> Menambah Pinjaman.....	69
Table 4.23 Skenario <i>Use Case</i> Mengubah Data Pinjaman .....	70
Table 4.24 Skenario <i>Use Case</i> Menghapus Data Pinjaman.....	70
Table 4.25 Skenario <i>Use Case</i> Melihat Data Anggota.....	71
Table 4.26 Skenario <i>Use Case</i> Penambahan Data Anggota .....	71
Table 4.27 Skenario <i>Use Case</i> Merubah Data Anggota .....	72
Table 4.28 Skenario <i>Use Case</i> Menghapus Data Anggota .....	73
Table 4.29 Skenario <i>Use Case Logout</i> .....	73
Table 4.30 Skenario <i>Use Case</i> Melihat Detail Simpanan (Tambahan dari Fase Elaborasi).....	74
Table 4.31 Skenario <i>Use Case</i> Melihat Detail Pinjaman (Tambahan dari Fase Elaborasi).....	74
Table 4.32 Skenario <i>Use Case</i> Menghitung Estimasi Pinjaman (Tambahan dari Fase Elaborasi).....	75
Table 4.33 Skenario <i>Use Case</i> Unduh Form Pengajuan (Tambahan dari Fase Elaborasi).....	75
Table 4.34 Skenario <i>Use Case</i> Edit Password (Tambahan dari Fase Elaborasi) ....	76
Table 4.35 Skenario <i>Use Case</i> Melihat Detail Pengajuan (Tambahan dari Fase Konstruksi).....	76
Table 5.1 Algoritma Fungsi Simpanan() .....	108
Table 5.2 Algoritma Fungsi Detail_simpanan() .....	109
Table 5.3 Algoritma Fungsi InsertPengajuan() .....	109
Table 5.4 Algoritma Fungsi DetailPengajuan() .....	109
Table 6.1 Spesifikasi Perangkat Keras .....	116
Table 6.2 Spesifikasi Perangkat Lunak .....	116
Table 6.3 Implementasi <i>Class</i> .....	117
Table 6.4 Kode Program Fungsi simpanan() .....	118

Table 6.5 Kode Program Fungsi detail_simpanan() .....	119
Table 6.6 Kode Program Fungsi insertPengajuan() .....	120
Table 6.7 Kode Program Fungsi detailPengajuan() .....	120
Table 7.1 Perancangan Pengujian <i>Black-Box</i> .....	127
Table 7.2 Table hasil pengujian <i>black box</i> .....	135
Table 7.3 Skenario <i>task</i> pengujian UAT .....	141
Table 7.4 Kriteria dan pertanyaan UAT .....	142
Table 7.5 Hasil Wawancara User Acceptance Testing .....	143



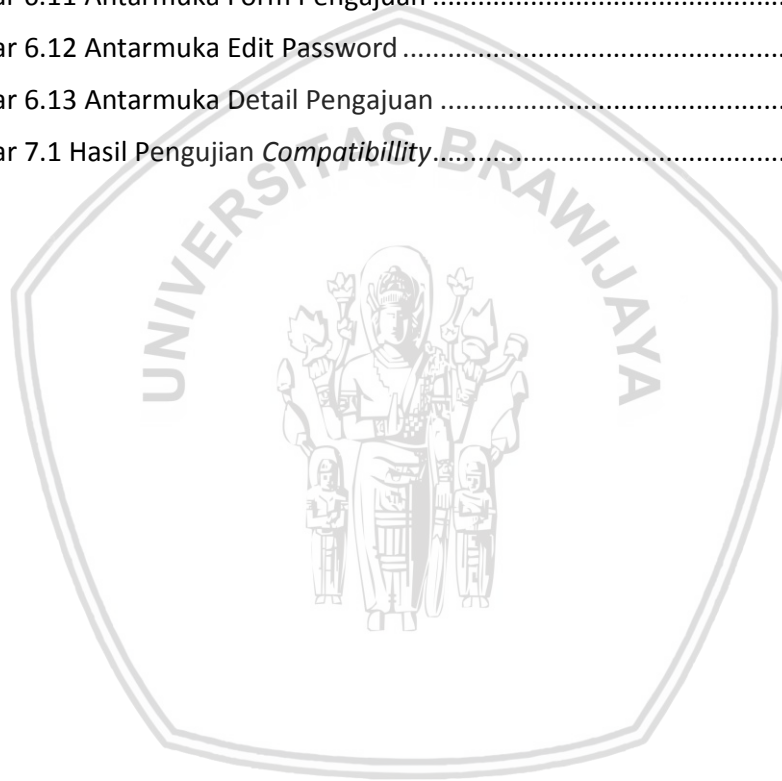
## DAFTAR GAMBAR

Gambar 2.1 Fase <i>Customer Relationship Management</i> .....	13
Gambar 2.2 Contoh <i>Physical Data Model</i> .....	29
Gambar 2.3 Skema <i>Model View Controller</i> .....	29
Gambar 2.4 Fase – Fase pada Metode RUP .....	31
Gambar 3.1 Diagram Penelitian .....	37
Gambar 4.1 As-Is mendapatkan informasi simpanan .....	42
Gambar 4.2 As-is mendapatkan informasi pinjaman .....	42
Gambar 4.3 As-Is pengajuan pinjaman .....	44
Gambar 4.4 As-Is setoran .....	45
Gambar 4.5 To-Be informasi pinjaman .....	48
Gambar 4.6 To-Be informasi simpanan .....	49
Gambar 4.7 To-Be setoran pinjaman .....	50
Gambar 4.8 To-Be pengajuan pinjaman .....	52
Gambar 4.9 Kodefikasi Fitur .....	57
Gambar 4.10 Kodefikasi Kebutuhan .....	58
Gambar 4.11 <i>Use Case Diagram</i> Fase Insepsi .....	62
Gambar 4.12 <i>Use Case</i> Fase Konstruksi .....	63
Gambar 4.13 <i>Activity diagram login</i> .....	78
Gambar 4.14 <i>Activity diagram</i> melihat info simpanan .....	79
Gambar 4.15 <i>Activity diagram</i> melihat informasi pinjaman .....	79
Gambar 4.16 <i>Activity Diagram</i> pengajuan pinjaman .....	80
Gambar 4.17 <i>Activity Diagram</i> konfirmasi pinjaman .....	81
Gambar 4.18 <i>Activity diagram</i> melihat data simpanan .....	82
Gambar 4.19 <i>Activity diagram</i> merubah data simpanan .....	83
Gambar 4.20 <i>Activity diagram</i> melihat data pinjaman .....	84
Gambar 4.21 <i>Activity diagram</i> tambah data pinjaman .....	85
Gambar 4.22 <i>Activity diagram</i> mengubah data pinjaman .....	86
Gambar 4.23 <i>Activity diagram</i> hapus data pinjaman .....	87
Gambar 4.24 <i>Activity diagram</i> melihat data anggota .....	88
Gambar 4.25 <i>Activity diagram</i> menambah data anggota .....	89
Gambar 4.26 <i>Activity diagram</i> mengubah data anggota .....	90

Gambar 4.27 <i>Activity diagram</i> menghapus data anggota .....	91
Gambar 4.28 <i>Activity diagram</i> logout .....	92
Gambar 4.29 <i>Activity diagram</i> melihat detail simpanan .....	93
Gambar 4.30 <i>Activity diagram</i> melihat detail pinjaman .....	94
Gambar 4.31 <i>Activity diagram</i> menghitung estimasi .....	95
Gambar 4.32 <i>Activity diagram</i> unduh file .....	96
Gambar 4.33 <i>Activity diagram</i> edit password .....	97
Gambar 4.34 <i>Activity diagram</i> melihat detail pengajuan .....	97
Gambar 5.1 <i>Sequence Diagram</i> Melihat Informasi Simpanan .....	99
Gambar 5.2 <i>Sequence Diagram</i> Melihat Detail Simpanan .....	100
Gambar 5.3 <i>Sequence Diagram</i> Melihat Informasi Pinjaman .....	101
Gambar 5.4 <i>Sequence Diagram</i> Melihat Detail Pinjaman .....	101
Gambar 5.5 <i>Sequence Diagram</i> Mengajukan Pinjaman .....	102
Gambar 5.6 <i>Sequence Diagram</i> Menghitung Estimasi .....	103
Gambar 5.7 <i>Sequence Diagram</i> Mengubah Password .....	104
Gambar 5.8 <i>Sequence Diagram</i> Mengunduh Form Pinjaman .....	105
Gambar 5.9 <i>Sequence Diagram</i> Melihat Detail Pengajuan .....	105
Gambar 5.10 <i>Class Diagram Model</i> .....	106
Gambar 5.11 <i>Class Diagram Controller</i> .....	107
Gambar 5.12 Perancangan Basis Data .....	108
Gambar 5.13 Antarmuka <i>Login</i> .....	110
Gambar 5.14 Antarmuka Beranda Anggota .....	111
Gambar 5.15 Antarmuka Beranda Anggota (literasi kedua fase elaborasi) .....	111
Gambar 5.16 Antarmuka Simpanan .....	112
Gambar 5.17 Antarmuka Detail Simpanan .....	112
Gambar 5.18 Antarmuka Pinjaman Anggota .....	113
Gambar 5.19 Antarmuka Detail Pinjaman .....	114
Gambar 5.20 Perancangan Antarmuka Pengajuan Pinjaman .....	114
Gambar 5.21 Perancangan Antarmuka Form Pengajuan Pinjaman .....	115
Gambar 5.22 Antarmuka Konfirmasi Pengajuan .....	115
Gambar 6.1 Implementasi Basis Data .....	118
Gambar 6.2 Antarmuka login anggota .....	121



Gambar 6.3 Antarmuka <i>Home</i> Anggota.....	121
Gambar 6.4 Antarmuka simpanan anggota.....	122
Gambar 6.5 Antarmuka Detail Simpanan .....	122
Gambar 6.6 Antarmuka Detail Simpanan (Fase Konstruksi).....	123
Gambar 6.7 Antarmuka pinjaman anggota.....	123
Gambar 6.8 Antarmuka detail pinjaman.....	124
Gambar 6.9 Antarmuka Pengajuan Pinjaman Fase Elaborasi.....	124
Gambar 6.10 Antarmuka Pengajuan Pinjaman Fase Konstruksi.....	125
Gambar 6.11 Antarmuka Form Pengajuan .....	125
Gambar 6.12 Antarmuka Edit Password .....	126
Gambar 6.13 Antarmuka Detail Pengajuan .....	126
Gambar 7.1 Hasil Pengujian <i>Compatibillity</i> .....	141



## BAB 1 PENDAHULUAN

### 1.1 Latar belakang

Kemajuan teknologi informasi berjalan dengan cepat dan mempengaruhi berbagai bidang dalam kehidupan. Teknologi memberikan kemudahan bagi setiap kegiatan manusia sehingga menjadi lebih efektif dan efisien. Kemajuan ini mendorong setiap organisasi menerapkan teknologi untuk membantu mencapai tujuan, visi dan misi organisasi. Informasi merupakan salah satu hal yang penting dalam suatu organisasi. Untuk menciptakan efisiensi informasi dalam organisasi diperlukan adanya manajemen sistem informasi.

Koperasi merupakan sebuah perkumpulan yang beranggotakan orang – orang atau badan hukum yang memberikan kebebasan masuk dan keluar sebagai anggota dengan bekerja sama secara kekeluargaan menjalankan usaha untuk meningkatkan kesejahteraan para anggotanya (Arifinal, 1984 disitasi dalam Sitio & Tamba, 2001). Koperasi berdasar pada asas kekeluargaan sehingga dalam operasionalnya koperasi harus terbuka terhadap setiap anggotanya. Salah satu jenis koperasi yang ada di Indonesia berdasarkan keanggotaannya adalah Koperasi Pegawai Republik Indonesia (KPRI) dimana status pegawai negeri yang menjadi syarat keanggotaannya. Pada umumnya KPRI didirikan pada lingkup instansi pemerintah yang bertujuan untuk meningkatkan kesejahteraan pegawai negeri.

KPRI “GURU” Kec. Sumbermanjing Wetan merupakan koperasi dibentuk pada lingkup instansi pendidikan yang mencakup area kecamatan dan dilegalisasi secara badan hukum pada 6 September 1996. Pada saat ini koperasi ini dikelola oleh 3 orang pengawas, 7 orang pengurus dan 6 orang pegawai yang melayani 426 anggota. Salah satu bidang usaha yang ditawarkan oleh KPRI kepada anggotanya adalah Unit Simpan Pinjam. Pada unit simpan pijam setiap anggota mempunyai hak dan kewajiban sesuai dengan peraturan khusus yang dibuat dan disepakati oleh seluruh anggota koperasi. Setiap anggota memiliki kewajiban untuk membayar simpanan wajib dimana hal tersebut yang menjadi sumber dana dari unit simpan pinjam. Sehingga loyalitas dari anggota sangat berpengaruh terhadap bertahannya sebuah koperasi. Sedangkan anggota juga mempunyai hak untuk memperoleh piutang atau pinjaman yang besarnya telah ditentukan melalui peraturan yang ada.

Saat ini pada unit simpan pijam KPRI “GURU” Kec. Sumbermanjing Wetan dikelola oleh 2 orang pegawai. Dalam kegiatan proses bisnis unit simpan pijam yang saat ini untuk mendapatkan informasi mengenai pengajuan pinjaman mengharuskan anggota mendatangi bagian administrasi. Pada proses pengajuan pinjaman, anggota akan melakukan konsultasi mengenai berapa jumlah maksimal pinjaman, kurun waktu, besaran bunga dan besaran cicilan kepada petugas sebelum memutuskan melakukan peminjaman. Namun dalam penerapannya masih terdapat beberapa kekurangan, permasalahan muncul antara lain berasal dari letak geografis Kecamatan Sumbermanjing Wetan sendiri yang mempunyai luas wilayah yang besar yaitu mencakup 14 desa dan memiliki kondisi topografi

Dari uraian diatas, penulis melakukan penelitian “*PENGEMBANGAN SISTEM INFORMASI CUSTOMER-TOUCHING APPLICATIONS UNIT SIMPAN PINJAM MENGGUNAKAN METODE RUP (STUDI PADA KPRI “GURU” KECAMATAN SUMBERMANJING WETAN)*”. Dengan adanya penelitian tersebut diharapkan dapat membuktikan bahwa pengembangan sistem *Customer-Touching Applications* dengan metode RUP dapat membuat sistem yang sesuai dengan kebutuhan dan mampu meningkatkan kegiatan proses bisnis dari KPRI “GURU” Kec. Sumbermanjing Wetan.

## 1.2 Rumusan masalah

Berdasarkan latar belakang diatas maka rumusan masalah yang dapat dikaji adalah sebagai berikut:

1. Bagaimana hasil analisa proses bisnis dan kebutuhan sistem *customer-touching applications* pada fase insepasi yang akan diterapkan dalam pengembangan sistem informasi unit simpan pinjam KPRI “GURU” Kec. Sumbermanjing Wetan menggunakan metode *Rational Unified Process (RUP)*?
2. Bagaimana hasil perancangan sistem pada fase elaborasi dalam pengembangan sistem informasi unit simpan pinjam KPRI “GURU” Kec. Sumbermanjing Wetan menggunakan metode *Rational Unified Proses (RUP)*?
3. Bagaimana hasil implementasi dan pengujian sistem pada fase konstruksi dalam pengembangan sistem informasi unit simpan pinjam KPRI “GURU” Kec. Sumbermanjing Wetan menggunakan metode *Rational Unified Process (RUP)*?
4. Bagaimana hasil pengujian *user acceptance testing* pada fase transisi dalam pengembangan sistem informasi unit simpan pinjam KPRI “GURU” Kec. Sumbermanjing Wetan menggunakan metode *Rational Unified Process (RUP)*?

## 1.3 Tujuan

Berdasarkan rumusan masalah yang telah dibuat, penelitian ini memiliki beberapa tujuan anantara lain:

1. Melakukan analisis proses bisnis dan kebutuhan sistem *customer-touching applications* pada fase insepasi dalam pengembangan sistem informasi unit simpan pinjam KPRI “GURU” Kec. Sumbermanjing Wetan menggunakan metode *Rational Unified Process (RUP)*.
2. Merancang sistem informasi pada fase elaborasi dalam pengembangan sistem informasi unit simpan pinjam KPRI “GURU” Kec. Sumbermanjing Wetan menggunakan metode *Rational Unified Process (RUP)*.
3. Mengimplementasikan dan menguji sistem pada fase konstruksi dalam pengembangan sistem informasi unit simpan pinjam KPRI “GURU” Kec.

Menguraikan tentang dasar teori yang digunakan mengenai pembangunan sistem informasi unit simpan pinjam menggunakan metode *Customer-Touching Applications* pada KPRI “GURU” Kec. Sumbermanjing Wetan.

### BAB III METODOLOGI PENELITIAN

Membahas tentang metode dan langkah kerja yang digunakan dalam penulisan penelitian ini.

### BAB IV ANALISIS

Membahas pemodelan sistem *Customer Relationship Management* (CRM) yang akan dikembangkan meliputi permodelan proses bisnis, analisis kebutuhan dan permodelan *use case*.

### BAB V PERANCANGAN

Membahas hasil perencanaan dalam pembangunan sistem informasi sesuai permasalahan dan metode yang digunakan meliputi perancangan *sequence diagram*, *class diagram*, antarmuka dan perancangan *database*.

### BAB VI IMPLEMENTASI

Membahas tentang implementasi pembuatan sistem informasi berdasarkan metode penelitian dan perancangan dari sistem yang telah dibuat.

### BAB VII PENGUJIAN

Membahas tentang pengukuran pengaruh solusi atau hasil implementasi terhadap permasalahan yang diuraikan pada rumusan masalah.

### BAB VIII PENUTUP

Membuat kesimpulan dari hasil analisis dan memberikan saran demi pengembangan lebih lanjut untuk pencapaian yang lebih baik.

## BAB 2 LANDASAN KEPUSTAKAAN

### 2.1 Kajian Pustaka

Kajian pustaka yang digunakan pada penelitian ini merujuk pada penelitian sebelumnya yang berjudul "*Evaluation of the Effects of e-CRM on Customer Loyalty (Case Study: Esfahan Branch's of Sepah Bank)*" yang dilakukan oleh Somayeh Salehi pada tahun 2015. Latar belakang dari penelitian tersebut adalah untuk mengeksplorasi dampak strategi e-CRM pada loyalitas pelanggan sebuah bank. Hal ini didasari dengan adanya fakta bahwa loyalitas pelanggan merupakan sebuah hal penting dalam sebuah organisasi komersil atau penyedia layanan, selain itu terdapatnya perubahan dan perkembangan teknologi yang semakin baik. Penelitian ini adalah penelitian *descriptive* dimana menggunakan metode *correlation survey*, dimana penelitian dilakukan dengan memberikan kuisisioner secara acak kepada Esfahan Branch's of Sepah Bank, Iran. Hasil dari penelitian tersebut adalah strategi CRM khususnya e-CRM merupakan sebuah strategi kuat yang muncul di lingkungan yang kompetitif seperti dunia perbankan. Implementasi e-CRM oleh bank akan mengarah pada peningkatan kepuasan pelanggan dan kepuasan pelanggan merupakan faktor penentu dalam loyalitas pelanggan.

Daris Arifin dan Johan Setiawan melakukan penelitian yang berjudul "Rancangan Aplikasi e-CRM pada PT Essensilindo Perdana" pada tahun 2005. Latar belakang dari penelitian ini adalah PT Essensilindo Perdana yang merupakan distributor biscuit khong guan dan monde merasa perlu memiliki sebuah e-CRM untuk meningkatkan loyalitas dan *profitable customer* yang dimiliki. Penelitian dilakukan dengan menggali informasi dari bagian pemasaran dan penjualan mengenai apa saja yang dibutuhkan oleh pelanggan. Secara keseluruhan penelitian bersifat analisi perancangan, dimana hasil penelitian hanya sampai pada tahapan rancangan sistem. Terdapat 3 fitur utama yang direkomendasikan oleh peneliti yaitu fitur buku tamu dan tanya jawab, belanja online, dan rekomendasi produk. Setelah dibuat sebuah rancangan sistem, rancangan tersebut dievaluasi yang tujuannya untuk mengetahui kepuasan calon pengguna terhadap sistem yang akan dibuat. Evaluasi dilakukan dengan metode kuisisioner yang di dalamnya terdapat 3 komponen uji yaitu tampilan sistem, kelengkapan fitur dan kemudahan memperoleh informasi. Hasil dari penelitian tersebut adalah e-CRM mampu menjadi sebuah solusi tepat untuk memenuhi kebutuhan pelanggan, fitur – fitur yang ditawarkan dapat mempermudah pelanggan mendapatkan informasi sehingga meningkatkan efisiensi dan loyalitas.

Sebuah penelitian yang berjudul "*Positive Impact of CRM Implementation to Improving The Services of Animal Polyclinics Customers*" oleh Edy Paryitno dan Novita Amylia pada tahun 2017. Penelitian dilakukan pada sebuah layanan kesehatan hewan yaitu UPT Poliklinik Hewan Yogyakarta. Tujuan dari penelitian ini adalah untuk mengetahui dampak positif penerapan aplikasi CRM pada sebuah poliklinik hewan. Dalam penelitian ini terbagi menjadi 3 tahap utama yaitu analisis



kebutuhan, perancangan, dan implementasi. Setelah tahapan implementasi dari sistem, akan dilakukan evaluasi terhadap penerapan CRM. Hasil akhir dari penelitian tersebut adalah CRM secara menyeluruh membantu dalam layanan, terdapat perbaikan waktu tunggu sebesar 82%, meningkatkan layanan sebesar 50%.

Ghaniyu Zulfata melakukan penelitian yang berjudul "*Pengembangan Electronic Customer Relationship Management (E-CRM) Pada Toko Sepatu Aneka Sport Malang Dengan Metode Kerangka Kerja Dynamic CRM*" yang dilakukan oleh pada tahun 2017. Penelitian berdasar pada kebutuhan organisasi untuk mempertahankan pelanggannya ditengah persaingan yang ketat. Tujuan dari penelitian ini adalah untuk mengembangkan sebuah aplikasi yang berdasar pada strategi CRM dengan menggunakan *framework of Dynamic CRM*. *Dynamic CRM* digunakan pada tahapan analisis kebutuhan fungsional. Pengujian sistem yang telah dibuat meliputi pengujian *white-box*, *black-box*, *compability* dan *user acceptance testing*. Hasil dari penelitian tersebut adalah semua kebutuhan sistem dapat dideskripsikan sesuai dengan 3 fase dalam *dynamic CRM* sehingga menghasilkan sebuah apliaksi dengan tingkat penerimaan pengguna mencapai 81,9%.

Pada penelitian lainnya yang berjudul "*Analisis dan Implementasi Perancangan Metode Rational Unifed Process pada layanan SDB dan Metode Pengujian Product Metric pada Bank Mandiri Cabang Palu Sam Ratulangi*" oleh Muhammad Taufik pada tahun 2015. Bank Mandiri yang merupakan bank salah satu bank terbesar di Indonesia masih menerapkan sebuah sistem yang bersifat manual untuk layanan *save deposit box* (SDB). Layanan SDB memiliki peran yang cukup vital karena dapat dijadikan sebagai tolak ukur pertumbuhan dana nasabah. Pengembangan menggunakan metode RUP dipilih untuk menciptakan sebuah sistem yang sesuai dengan kebutuhan. Dalam pengembangan sistem hanya menerapkan 3 dari total 4 fase RUP dan pengujian sistem menggunakan metode *product matric*. Hasil dari penelitian tersebut adalah metode RUP menciptakan sistem yang sesuai dengan kebutuhan dan berkualitas dengan pengujian yang baik.

Ardhani Dwi Kinasih dalam penelitian yang berjudul "*Pengembangan Sistem Evaluasi Pembelajaran PAUD (Studi Kasus di PAUD Seruni 05 Kota Malang)*" tahun 2017. Landasan dari penelitian ini adalah mengembangkan sebuah sistem evaluasi pembelajaran PAUD untuk menyelaraskan dengan tuntutan kurikulum 2013. Dalam pengembangan sistem tersebut peneliti menggunakan metode *Rational Unifed Process (RUP)* karena dapat mengakomodasi perubahan kebutuhan namun tidak berakibat pada penundaan waktu proyek. Metode RUP diterapkan secara menyeluruh meliputi keempat fasenya. Pengujian sistem menggunakan beberapa metode, kebutuhan fungsional menggunakan *unit testing* dan *integration testing* sedangkan *compatibility testing* untuk kebutuhan non fungsional. Hasil dari penelitian ini adalah RUP membantu pengembangan sistem yang baik ditandai dengan hasil pengujian *cyclomatic complexity* kurang dari 10, skor *SUS* sebesar 84,9 dan *compatibility* yang menunjukkan sistem dapat berjalan pada semua perangkat *Android* diatas versi 2.3.



Table 2.1 Kajian Pustaka

No	Judul Penelitian	Tujuan Penelitian	Metode	Kesimpulan
1	Somayeh Salehi, <i>Evaluation of the Effects of e-CRM on Customer Loyalty (Case Study: Esfahan Branch's of Sepah Bank)</i> . IEEE 2015	Mengetahui dampak strategi e-CRM terhadap loyalitas pelanggan pada organisasi perbankan.	<i>Correlation Survey, Customer Relationship Management</i>	Implementasi e-CRM oleh bank akan mengarah pada peningkatan kepuasan pelanggan dan loyalitas pelanggan.
2	Daris Arifin, "Rancangan Aplikasi e-CRM pada PT Essensiindo Perdana". <i>Soft Computing, Intelligent Systems and Information Technology</i> 2005.	Melakukan analisis dan perancangan aplikasi e-CRM berbasis website untuk meningkatkan loyalitas dan <i>profitable customer</i> .	<i>Customer Relationship Management</i>	Kemudahan yang diberikan oleh aplikasi e-CRM untuk mendapatkan informasi memberikan efisiensi dan layanan terhadap pelanggan.
3	Edy Paryitno dan Novita Amylia, "Positive Impact of CRM Implementation to Improving The Services of Animal Polyclinics Customers". IEEE. 2017	Mengetahui dampak positif penerapan strategi CRM pada layanan poliklinik hewan.	<i>Customer Relationship Management</i>	Penerapan CRM memberikan dampak berupa perbaikan waktu tunggu sebesar 82% dan meningkatkan layanan sebesar 50%.
4	Ghaniyu Zulfata, "Pengembangan Electronic Customer Relationship Management (E-CRM) Pada Toko Sepatu Aneka Sport Malang Dengan Metode Kerangka Kerja Dynamic CRM". J-PTIUK 2017	Melakukan analisis dan pengembangan sebuah sistem e-CRM untuk menghadapi persaingan dan mempertahankan pelanggan.	<i>Customer Relationship Management, framework of Dynamic CRM, Compatibility</i>	Pengembangan sistem e-CRM dapat diterima oleh calon penggunaanya dengan nilai sebesar 81,9%

No	Judul Penelitian	Tujuan Penelitian	Metode	Kesimpulan
			Testing, User Acceptance Testing	
5	Muhammad Taufik, "Analisis dan Implementasi Perancangan Metode Rational Unified Process pada Layanan SDB dan Metode Pengujian Product Metric pada Bank Mandiri Cabang Palu Sam Ratulangi". e-Proceeding of Engineering telkomuniversity.ac.id. 2015	Melakukan analisis dan pengembangan sistem save deposit box (SDB) menggunakan metode Rational Unified Process (RUP) pada Bank Mandiri.	Rational Unified Process (RUP), Product Matrix	Penggunaan metode RUP dapat membantu dalam pengembangan sistem yang sesuai dengan kebutuhan dan memiliki kualitas yang baik.
6	Ardhani Dwi Kinasih, "Pengembangan Sistem Evaluasi Pembelajaran PAUD (Studi Kasus di PAUD Seruni 05 Kota Malang)". J-PTIHK. 2017	Melakukan analisis dan pengembangan sistem evaluasi pembelajaran PAUD menggunakan metode Rational Unified Process (RUP).	Rational Unified Process (RUP), Unit Testing, Integration Testing, Compatibility Testing	Metode RUP dapat mengakomodasi perubahan yang ada pada proses pengembangan sistem sehingga dapat menghasilkan sebuah sistem yang sesuai dengan kebutuhan.

Berdasarkan penelitian – penelitian diatas terdapat keterkaitan dengan penelitian yang akan dilakukan. Penelitian dengan judul *Evaluation of the Effects of e-CRM on Customer Loyalty (Case Study: Esfahan Branch's of Sepah Bank)*, Rancangan Aplikasi e-CRM pada PT Essensilindo Perdana, dan *Positive Impact of CRM Implementation to Improving The Services of Animal Polyclinics Customers* memiliki keterkaitan dimana penelitian yang akan dilakukan menggunakan strategi CRM untuk meningkatkan layanan dan loyalitas pelanggan. Sedangkan penelitian yang berjudul “Pengembangan *Electronic Customer Relationship Management (E-CRM)* Pada Toko Sepatu Aneka Sport Malang Dengan Metode Kerangka Kerja Dynamic CRM” memiliki korelasi yaitu penggunaan metode pengujian UAT untuk menguji sistem dan strategi CRM yang akan diterapkan. Dua penelitian akhir yang berjudul “Analisis dan Implementasi Perancangan Metode *Rational Unified Process* pada layanan SDB dan Metode Pengujian *Product Metric* pada Bank Mandiri Cabang Palu Sam Ratulangi” dan “Pengembangan Sistem Evaluasi Pembelajaran PAUD (Studi Kasus di PAUD Seruni 05 Kota Malang)” memiliki keterkaitan pada penggunaan metode RUP untuk mengembangkan sistem dengan baik.

## 2.2 Koperasi

Koperasi berasal dari kata latin “*coopere*” atau dalam bahasa inggris “*cooperation*”. Sehingga secara etimologi koperasi mempunyai arti “kerja sama” atau sebuah kegiatan yang dilakukan secara bersama – sama dan mempunyai tujuan yang sama. Koperasi juga dapat diartikan sebagai suatu perkumpulan yang beranggotakan perseorangan atau badan hukum yang memberikan kebebasan kepada anggotanya untuk masuk atau keluar. Setiap anggota bekerja sama secara kekeluargaan menjalankan usaha untuk memperbaiki kesejahteraan jasmaniah anggotanya (Sitio & Tamba, 2001).

Secara internasional oleh *ILO (International Labour Organization)* mendefinisikan koperasi sebagai perkumpulan orang yang secara suka rela bergabung untuk bekerja sama dalam mendapatkan keuntungan dan menanggung resiko bersama. Dari definisi tersebut terdapat 6 elemen dasar dari koperasi yaitu perkumpulan, keanggotaan sukarela, adanya tujuan ekonomi, suatu organisasi bisnis, keadilan dan keseimbangan.

Bapak koperasi Indonesia yaitu Moh. Hatta mendefinisikan koperasi sebagai usaha bersama untuk memperbaiki perekonomian berdasarkan semangat tolong menolong. Indonesia sendiri mengatur organisasi koperasi dalam UU no. 25 tahun 1992 tentang Perkoperasian. Dalam undang – undang tersebut terdapat tujuan koperasi secara umum yaitu memajukan kesejahteraan anggota dan masyarakat pada umumnya serta ikut memanjudkan perekonomian nasional.

## 2.3 KPRI “GURU” Kecamatan Sumbermanjing Wetan

KPRI atau Koperasi Pegawai Republik Indonesia sebelumnya bernama Koperasi Pegawai Negeri (KPN) sudah menjadi wadah bagi pegawai negeri khususnya bagi guru di Kecamatan Sumbermanjing Wetan sejak tahun 1968.

Koperasi KPRI “GURU” Sumbermanjing Wetan mempunyai area yang luas terdiri dari 63 instansi pendidikan yang tergabung di dalamnya dengan jumlah anggota lebih dari 426 orang. Cakupan wilayah KPRI seluas 27.218,49 Ha dan terbagi menjadi 14 kecamatan dengan kondisi wilayah yang berbukit.

KPRI mempunyai mempunyai beberapa jenis unit antara lain unit simpan pinjam, unit toko dan unit transportasi. Unit simpan pinjam merupakan unit yang menyediakan layanan bagi anggota berupa jasa simpan atau tabung dan jasa pinjam atau kredit. Unit toko merupakan salah satu usaha KPRI yaitu menyediakan peralatan tulis sedangkan unit transportasi memberikan jasa penyewaan kendaraan. Berdasarkan pada asas keterbukaan dan kekeluargaan yang diterapkan oleh koperasi maka segala informasi mengenai koperasi dapat didapatkan oleh seluruh anggota. Termasuk dalam penetapan pengurus, setiap tahunnya diadakan pemilihan yang pencalonannya dari anggota dan dipilih oleh anggota.

## 2.4 Simpan Pinjam

Kegiatan simpan pinjam dilakukan oleh unit simpan pinjam yang didirikan dengan tujuan untuk meningkatkan kesejahteraan anggota koperasi, serta memiliki peran aktif dalam upaya meningkatkan kualitas kehidupan ekonomi (Sitio & Tamba, 2001). Usaha simpan pinjam merupakan salah jenis usaha yang telah dikenal secara luas oleh anggota koperasi dan masyarakat di Indonesia. Usaha ini adalah salah satu usaha lembaga keuangan non bank dilakukan untuk menghimpun dana dan menyalurkannya dari dan untuk anggotanya.

Pada jenis usaha ini dua faktor utamanya adalah simpanan (tabungan) dan pinjaman (piutang) itu sendiri. Pinjaman merupakan semua jenis pinjaman yang harus dibayar kembali bersama bunganya oleh peminjam sesuai dengan perjanjian yang telah disepakati. Pinjaman juga dapat didefinisikan sebagai bentuk kepercayaan memberikan atau menitipkan baik berupa barang, jasa, atau uang dari suatu pihak ke pihak lainnya, dimana pihak yang menerima harus mengembalikan dan memberikan balas jasa atas hal tersebut (Sitio & Tamba, 2001).

Sedangkan simpan atau tabungan adalah sebagian pendapatan yang diterima seseorang tidak semua digunakan untuk konsumsi, melainkan sebagian akan disimpan sebagai simpanan (Sitio & Tamba, 2001). Dalam koperasi simpan juga dapat berarti sebagai iuran yang telah disepakati. Terdapat berbagai jenis simpanan dalam sebuah koperasi yang memiliki fungsi yang berbeda-beda. Jenis – jenis simpanan dalam koperasi simpan pinjam antara lain simpanan pokok, wajib, berjangka dan lain sebagainya yang dapat ditentukan berdasarkan pada kesepakatan antar anggotanya.

## 2.5 Customer Relationship Management (CRM)

*Customer Relationship Management (CRM)* merupakan proses pengelolaan semua aspek interaksi yang dimiliki oleh perusahaan dengan

pelanggannya. Aspek – aspek tersebut meliputi prospeksi, penjualan dan layanan. CRM membantu untuk memberikan wawasan dan memperbaiki hubungan perusahaan dengan menggabungkan semua prespektif (Buttle, 2009).

CRM juga dapat di artikan sebagai sebuah strategi pendekatan untuk mengidentifikasi, mengakusisi dan mempertahankan pelanggan. Secara tidak langsung CRM dapat membantu untuk memaksimalkan setiap interaksi pelanggan. Hal ini dikarenakan CRM merupakan sebuah strategi bisnis inti yang mengintegrasikan proses dan fungsi internal dan eksternal untuk menciptakan nilai bagi pelanggan berdasarkan pada data berkualitas tinggi yang dimungkinkan oleh teknologi informasi.

Konsekuensi dari perkembangan dunia bisnis adalah setiap organisasi harus bersaing baik di pasar domestik maupun global. Cara untuk memenangkan persaingan, bisa dilakukan dengan cara mempererat hubungan antara pemangku kepentingan, pelanggan, distributor, karyawan dan lingkungan sekitar lainnya. Perusahaan harus merubah orientasi transaksi atau hubungan jangka pendek menjadi membangun hubungan jangka panjang (Payne, 2005).

### **2.5.1 Fase *Customer Relationship Management* (CRM)**

Berdasarkan strategi untuk mengelola pelanggan, CRM dibagi menjadi tiga fase, yaitu: (Kalakota & Robinson, 2001)

#### **1. *Acquire* (Mendapatkan pelanggan)**

Fase awal dalam hubungan dengan pelanggan adalah mendapatkan kepercayaan pelanggan itu sendiri, untuk mendapatkan pelanggan baru organisasi harus mempromosikan produk atau jasa yang inovatif. Selain itu organisasi harus memberikan gambaran bagaimana mereka memperlakukan pelanggan dengan baik sehingga dapat menarik hati calon pelanggannya. Faktor penting lainnya adalah menawarkan produk atau jasa yang lebih unggul dari kompetitor.

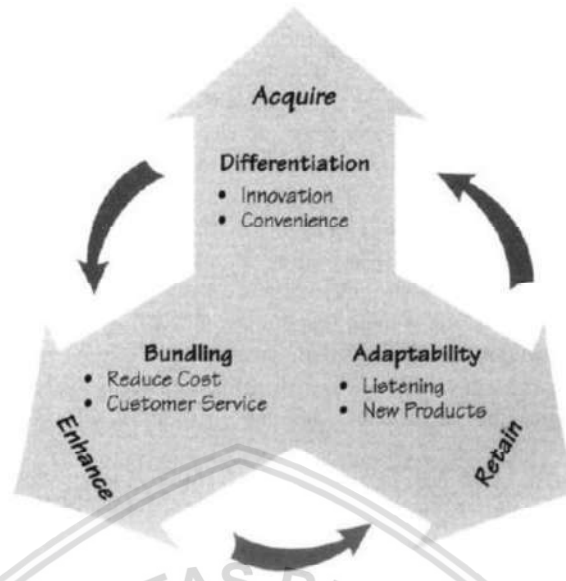
#### **2. *Enhance* (Meningkatkan keuntungan dari pelanggan yang sudah ada)**

Organisasi dituntut untuk mampu menciptakan hubungan yang erat dengan para pelanggan. Pada fase ini organisasi harus mendengarkan berbagai keluhan dan meningkatkan kualitas pelayanan dari pelanggan yang telah ada. Selain itu organisasi juga dapat menerapkan strategi *cross-selling* atau *up-selling* yang dapat meningkatkan pendapatan dan mengurangi biaya (*reduce cost*).

#### **3. *Retain* (Mempertahankan pelanggan)**

Mempertahankan pelanggan membutuhkan pengertian yang baik tentang apa yang dibutuhkan oleh pelanggan. Pada fase ini fokus utama organisasi ada pada pelayanan, organisasi dapat memenuhi apa yang dibutuhkan oleh pelanggan. Menerima segala keluhan atau saran dari pelanggan agar dijadikan sebagai bahan evaluasi untuk meningkatkan kualitas pelayanan yang ada serta memberikan kemudahan – kemudahan untuk memanjakan pelanggan.





**Gambar 2.1 Fase Customer Relationship Management**

Sumber : Kalakota & Robinson (2001)

### 2.5.2 Jenis Customer Relationship Management (CRM)

Terdapat empat tipe CRM yang mendukung kerjanya strategi CRM yaitu (Buttle, 2009):

a. *Strategic CRM*

*Strategic CRM* memiliki fokus utama untuk mempertahankan pelanggan yang loyal. Dengan cara memberikan *value* atau nilai yang lebih baik dibandingkan dengan para pesaing. Pendekatan yang sering digunakan pada *strategic CRM* antara lain pendekatan terhadap produk, produksi dan penjualan.

b. *Operasional CRM*

*Operasional CRM* memiliki fokus utama kepada bagaimana untuk mengotomatisasi dan meningkatkan proses bisnis yang ada untuk mendukung pelayanan terhadap pelanggan. Fungsi aplikasi CRM ini dapat diterapkan pada pemasaran, penjualan maupun layanan. Dalam *operasional CRM* terdapat 3 jenis aplikasi utama yaitu *marketing automation*, *sales force automation*, dan *service automation*.

c. *Analytical CRM*

*Analytical CRM* memiliki fokus untuk menyimpan dan mengolah data untuk mendapatkan data yang berkaitan dengan peningkatan nilai pelanggan. Dasar dari analytical CRM adalah data dari pelanggan seperti data pembelian, pembayaran dan jasa yang digunakan, selain itu juga dikombinasikan dengan data eksternal perusahaan seperti gaya hidup dan demografi. Tujuan utamanya adalah untuk menganalisa pelanggan mana yang layak untuk dipertahankan.



d. *Collaborative CRM*

*Collaborative CRM* berfokus dalam penggunaan aplikasi CRM untuk berkomunikasi melampaui batas organisasi. Penggunaan teknologi di seluruh jajaran organisasi ini dengan tujuan untuk mengoptimalkan nilai perusahaan, partner dan pelanggan. Dengan penggunaan *collaborative CRM*, memungkinkan untuk menangani perbedaan jenis aplikasi dipakai oleh pelanggan, seperti *email*, telepon, *fax* dan *website*.

## 2.6 Operational CRM

*Operasional Customer Relationship Management (CRM)* lebih terfokus pada otomatisasi cara perusahaan dalam hubungan dengan pelanggannya. Berbagai aplikasi perangkat lunak *Customer Relationship Management (CRM)* memungkinkan untuk mengubah fungsi pemasaran, penjualan, dan pelayanan dapat berjalan secara otomatis. Beberapa aplikasi terpenting yang dapat diterapkan dalam operasional Customer Relationship Management (CRM), yaitu (Buttle, 2009):

a. *Marketing Automation (MA)*

*Marketing automation* menerapkan teknologi untuk mendukung pemasaran secara otomatis. Proses otomatisasi berdasar pada data pelanggan untuk mengembangkan, melaksanakan dan mengevaluasi komunikasi dan penawaran. Manfaat dari *marketing automation* saat ini ketika banyak sekali *platform* pemasaran yang mampu menarik pelanggan menjadi lebih mudah untuk dilakukan.

b. *Sales Force Automation (SFA)*

*Sales force automation* menerapkan teknologi untuk membantu perusahaan dalam kegiatan penjualan. Software SPA dapat dikonfigurasi sehingga pada proses penjualan dapat diaplikasikan pada banyak industri atau organisasi. SFA dapat membantu perusahaan karena perusahaan secara otomatis dapat menetapkan prospek dan melacak peluang.

c. *Service Automation (SA)*

*Service automation* adalah penerap aplikasi teknologi komputer untuk mendukung layanan perusahaan baik layanan langsung maupun layanan pendukung. Penggunaan *service automation* memungkinkan perusahaan untuk mengelola pelayanan kepada pelanggan secara otomatis. Otomatisasi layanan biasanya digunakan untuk mengatur dan menampung keluhan pelanggan yang mempunyai masalah dengan sebuah produk atau layanan.

Pada operasional CRM selain dibagi menjadi 3 jenis di atas terdapat 2 tipe aplikasi yaitu: (Rainer, 2010).

a. *Customer facing application*

*Customer facing application* merupakan aplikasi operasional CRM yang dibuat untuk mendukung para staf perusahaan atau organisasi dalam

melayani pelanggan. Fungsi dari aplikasi ini lebih bersifat sebagai alat bantu ketika menghadapi atau memberikan pelayanan terhadap pelanggan. Penggunaan tipe aplikasi ini memberikan kemudahan serta meningkatkan kualitas dari layanan yang diberikan.

*b. Customer touching application*

*Customer touching application* merupakan aplikasi yang langsung berinteraksi dengan pelanggan. Aplikasi itu dapat bekerja apabila ada permintaan layanan dari pelanggan, otomatis secara periodik, atau jika *event* tertentu dari perusahaan. Sehingga dengan adanya aplikasi ini dapat menggantikan perwakilan atau staf perusahaan untuk berinteraksi dengan pelanggan.

### **2.6.1 Service Automation (SA)**

*Service Automation (SA)* atau otomatisasi layanan memungkinkan perusahaan untuk mengotomatisasi fungsi layanan perusahaan terhadap pelanggan melalui *call center*, *contact person*, *helpdesk*, layanan berbasis web atau bertatap muka langsung dengan pelanggan di lapangan. *Service Automation (SA)* juga menerapkan teknologi informasi internet khususnya web, melalui web suatu perusahaan dapat memberikan pelayanan sesuai keinginan pelanggan. Manfaat *Service Automation (SA)* bagi perusahaan dan pelanggan yaitu pelayanan kepada pelanggan lebih efektif dan efisien, mengurangi biaya layanan, meningkatkan kualitas layanan, dan meningkatkan produktivitas layanan ke pelanggan.

Adapun macam-macam aktivitas pada modul *Service Automation (SA)* yaitu aktivitas manajemen, agen manajemen, kasus tugas, manajemen kasus, manajemen kontrak, *customer self-service*, *email respon* manajemen, eskalasi, *inbound* komunikasi manajemen, *outbound* komunikasi manajemen, antrian dan *routing*, penjadwalan, *scripting*, analisis layanan, layanan manajemen tingkat, manajemen *resource*, web kolaborasi, dan alur kerja teknik (Buttle, 2009). Sedangkan *Customer Self-Service* adalah layanan berbasis dimana pelanggan dapat melayani dirinya sendiri untuk melakukan transaksi, pemesanan, melaporkan pemesanan secara online dengan fasilitas web yang telah disediakan perusahaan sesuai dengan kebutuhan pelanggan (Adrian, 2005).

### **2.6.2 Customer-Touching Applications**

*Customer Touching Applications* merupakan salah satu bagian dari CRM. Pada operasional CRM salah satu bagian penting adalah *customer touching applications* karena merupakan sebuah aplikasi yang mampu membantu pemberian layanan secara otomatis. *Customer Touching Applications* sendiri sering disebut sebagian *E-CRM* atau pendekatan CRM yang memanfaatkan teknologi untuk berinteraksi kepada pelanggannya guna memberikan pelayanan yang lebih baik. *Customer Touching Applications* diartikan teknologi atau aplikasi yang mampu membuat pelanggan untuk langsung berinteraksi dengan organisasi. Dengan aplikasi tersebut pelanggan bias membantu kebutuhan dirinya sendiri (Raine, 2010)

## 2.7 Proses Bisnis

Proses bisnis merupakan sebuah kumpulan aktivitas dimana di dalamnya terdapat satu atau lebih *input* dan menghasilkan sebuah *output* seperti laporan yang nantinya bernilai bagi pelanggan (Monk & Wagner, 2013). Pelanggan atau *customer* dalam hal ini terbagi menjadi 2 yaitu *internal customer* dan *external customer*. *Internal customer* didefinisikan sebagai orang yang membeli atau menggunakan produk akhir, sedangkan *external customer* merupakan orang dari bagian lain dalam sebuah organisasi yang sama.

Dalam sebuah organisasi proses bisnis merupakan inti dari desain organisasi yang berorientasi pada proses. Berbeda dengan struktur organisasi yang membagi sebuah organisasi secara parsial tetapi proses bisnis lebih berkaitan dengan pelaksanaan tugas – tugas serta koordinasi dari beberapa aspek (Becker, et al., 2013). Contoh dari sebuah proses bisnis adalah pemesanan pada pabrik, proses penjualan *retail* dan persetujuan kredit pada bank.

### 2.7.1 Permodelan proses bisnis *as-is*

Permodelan proses bisnis *as-is* adalah langkah untuk mengumpulkan dan menggambarkan proses bisnis aktual pada sebuah organisasi. Permodelan ini tidak hanya berfungsi untuk menggambarkan proses bisnis yang saat ini, tetapi juga untuk mengetahui masalah atau kekurangan dari proses bisnis sehingga dapat memunculkan gagasan untuk meningkatkan sebuah proses bisnis dalam sebuah organisasi (Becker, et al., 2013).

Pada permodelan proses bisnis *as-is* informasi yang digunakan untuk menggambarkan kondisi aktual dari sebuah proses bisnis sebuah organisasi dapat menggunakan metode observasi atau wawancara. Observasi dilakukan untuk mengetahui secara nyata apa saja yang terjadi dalam sebuah proses bisnis. Sedangkan wawancara dapat dilakukan untuk mengetahui informasi yang lebih detail (Przybytek, 2011).

### 2.7.2 Permodelan proses bisnis *to-be*

Permodelan proses bisnis *to-be* merupakan sebuah permodelan untuk menggambarkan hasil *improvement* dari proses bisnis yang ada saat ini (*as-is*). Proses bisnis *to-be* harus dibuat berdasarkan pada proses bisnis *as-is* dan harus disetujui oleh pemangku kepentingan. Dalam penerapannya perubahan proses bisnis *as-is* ke *to-be* dapat dalam jangka waktu yang singkat atau lama (Becker, et al., 2013).

Pada permodelan bisnis *to-be* peran pemangku kepentingan sangat penting hal ini dikarenakan dalam perbaikan proses bisnis pemangku kepentingan dapat mengetahui keadaan dari organisasinya. Terdapat dua langkah utama dalam proses permodelan *to-be*, yang pertama adalah melakukan identifikasi peluang dari perbaikan yang ada. Hal ini untuk mencari proses yang dapat dikurangi atau diotomatisasi. Langkah selanjutnya adalah menentukan mana proses bisnis dasar

dan sejauh mana kita dapat melakukan otomatisasi dari proses yang ada (Przybytek, 2011).

## 2.8 Business Process Modelling Notation (BPMN)

*Business Process Modelling Notation* (BPMN) adalah grafis untuk menggambarkan langkah – langkah logis proses bisnis beserta aktor yang terlibat didalamnya. *Business Process Modelling Notation* (BPMN) juga merupakan sebuah standart notasi yang telah disepakati untuk menggambarkan proses bisnis. BPMN bertujuan untuk memberi notasi yang dapat dipahami oleh semua pengguna bisnis, mulai dari analisis bisnis yang membuat rancangan, implementator, serta orang bisnis yang mengelola dan mengontrol proses tersebut (Silver, 2012).


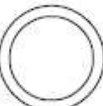

Didalam BPMN terdapat *Business Process Diagram (BPD)* yang berfungsi untuk menggambarkan proses bisnis. BPD pada dasarnya sama dengan *flowchart*, yang dirancang untuk menyajikan urutan dari semua aktifitas yang terjadi selama suatu proses. BPD merupakan diagram yang dirancang untuk membantu analisis dalam merancang, mengendalikan dan mengelola proses.

### 2.8.1 Flow Objects

#### 1. Event

*Event* digambarkan dalam bentuk lingkaran yang merupakan suatu hal yang dapat mempengaruhi proses bisnis. Terdapat tiga jenis *event* yaitu *start event*, *intermediate event*, dan *end event* yang dibedakan berdasarkan aliran prosesnya. Dari ketiga jenis *event* ini dapat menangkap (*catch*) atau melempar (*throw*) *event*. (Object Management Group, 2011)

**Table 2.2 Tipe Event dalam BPMN**


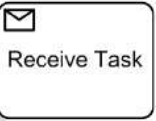
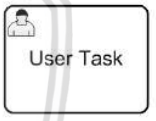

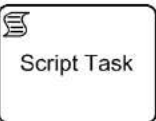
No	Tipe	Diskripsi	Simbol
1	<i>Start</i>	Mendeskripsikan dimana awal dari suatu proses.	
2	<i>Intermediate</i>	Mendeskripsikan dimana sesuatu terjadi diantara awal dan akhir proses. Dimana hal itu akan mempengaruhi alur dari proses, tapi tidak akan memulai atau menghentikan proses tersebut.	
3	<i>End</i>	Mendeskripsikan suatu akhir dari sebuah proses	

Sumber : Object Management Group (2011)

## 2. Activity

*Activity* digambarkan dalam bentuk persegi dengan ujung yang membulat. Merupakan sebuah aktivitas yang ada dalam sebuah rangkaian proses bisnis. Dalam BPD activity dibagi menjadi beberapa jenis menurut jenis dari aktifitas itu sendiri. (Object Management Group, 2011)

**Table 2.3 Tipe Activity dalam BPMN**

No	Tipe	Diskripsi	Simbol
1	<i>Service Task</i>	<i>Service task</i> adalah aktivitas yang dikerjakan oleh perangkat lunak	
2	<i>Receive Task</i>	<i>Receive task</i> adalah sebuah aktivitas menerima pesan dari aktivitas lain. <i>Task</i> ini merupakan alternatif dari catching message event.	
3	<i>User Task</i>	<i>User task</i> dikerjakan oleh manusia. Berbeda dengan manual task setelah user task diselesaikan, engine menunggu hasil dari pekerjaan atau aktivitas yang diselesaikan untuk kemudian diproses.	
4	<i>Manual Task</i>	<i>Task</i> ini dieksekusi oleh manusia secara manual yang tidak mempengaruhi aktivitas yang sudah diselesaikan oleh <i>engine</i> pemrosesan	
5	<i>Script Task</i>	<i>Task</i> ini mengeksekusi <i>script</i> yang sudah didefinisikan dan bisa dijalankan oleh <i>engine</i> pemrosesan. Saat <i>script</i> selesai maka tugas juga selesai.	

Sumber : BPMN 2.0 Handbook Second Edition (2012)

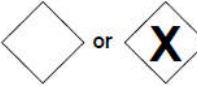




## 3. Gateway

*Gateway* digambarkan dengan bentuk berlian dan berfungsi untuk mengontrol alur proses yang berlangsung. *Gateway* mengendalikan



pertemuan dan perpisahan pada sebuah alur proses. Pada gateway banyak *input* atau *output* dapat dipengaruhi oleh jenis *gateway* masing – masing. Pada BPD terdapat beberapa jenis dari yang menunjukkan fungsi sebagai berikut :

**Table 2.4 Tipe Gateway dalam BPMN**

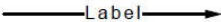
No	Tipe	Diskripsi	Simbol
1	<i>Exclusive</i>	Digunakan untuk membuat jalur dengan mengevaluasi keadaan kemudian mengarahkan kepada satu jalur yang sesuai	 or 
2	<i>Parallel</i>	Digunakan untuk melakukan percabangan proses dimana alur yang ada dijalankan secara bersamaan tanpa melihat proses sebelumnya.	
3	<i>Inclusive</i>	Digunakan ketika terdapat percabangan jalur dimana terdapat kondisi tertentu yang harus dipenuhi sehingga proses dapat diteruskan ke beberapa jalur.	
5	<i>Event-Based</i>	Digunakan ketika terdapat percabangan dimana bukan berdasarkan kondisi apa yang telah dipenuhi melainkan kondisi apa yang telah terjadi.	

Sumber : [www.lucidchart.com](http://www.lucidchart.com)



### 2.8.2 Connection

*Connecting object* merupakan penghubung antar *flow object* satu dengan yang lainnya atau elemen - elemen lainnya. *Flow object* yang saling terhubung akan membentuk struktur dasar dari BPD. Terdapat tiga jenis *connecting flow* pada BPMN, yaitu *message flow*, *sequence flow* dan *association flow*.

**Table 2.5 Tipe Connection dalam BPMN**

No	Tipe	Diskripsi	Simbol
1	<i>Sequence flow</i>	Digunakan untuk menunjukkan sebuah urutan kegiatan yang akan dilakukan	



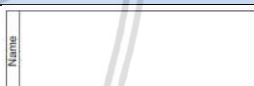

		dapat berupa aliran kondisional maupun <i>default</i>	
2	<i>Message flow</i>	Digunakan untuk menunjukkan pesan yang melintasi " <i>pool</i> " atau batasan organisasi.	
3	<i>Association flow</i>	Digunakan untuk menunjukkan hubungan antara <i>artifact</i> dengan aktifitas.	

Sumber : Object Management Group (2011)

### 2.8.3 Swimlanes

*Swimlanes* merupakan mekanisme untuk menggambarkan pembagian suatu tanggung jawab dari komponen atau aktor yang terlibat dalam sebuah proses bisnis. Ada yang *swimlanes* berfungsi untuk memperjelas penanggung jawab dari tiap aktifitas yang ada. Dalam BPMN terdapat 2 bentuk atau jenis dari *swimlanes* yaitu *lanes* dan *pool*.

Table 2.6 Tipe *Swimlanes* dalam BPMN

No	Tipe	Diskripsi	Simbol
1	<i>Pool</i>	<i>Pool</i> mempresentasikan tanggung jawab untuk aktivitas secara garis besar. <i>Pool</i> biasanya digunakan untuk menggambarkan sebuah organisasi.	
2	<i>Lanes</i>	<i>Lanes</i> merupakan sub bagian dari <i>pool</i> . <i>Lane</i> membagi <i>pool</i> menjadi sebuah hirarki sehingga dapat menggambarkan tanggung jawab secara lebih rinci seperti pembagian dalam sebuah organisasi	

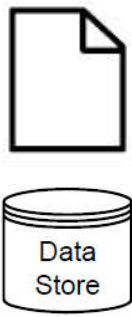
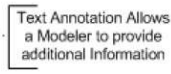

Sumber : Object Management Group (2011)

### 2.8.4 Artifact

*Artifact* merupakan informasi tambahan untuk memperjelas suatu proses bisnis. Dalam proses bisnis tidak mempengaruhi alur atau aktifitas yang ada. Dalam BPMN terdapat tiga jenis *artifact* yaitu *data object*, *group* dan *annotation*.

Table 2.7 Tipe *Artifact* dalam BPMN

No	Tipe	Diskripsi	Simbol
----	------	-----------	--------

1	<i>Data Object</i>	<i>Data</i> memrepresentasikan data berupa dokumen, e-mail atau surat. Selain itu terdapat <i>data store</i> merupakan tempat dimana proses dapat membaca atau menulis data.	
2	<i>Annotation</i>	Merupakan informasi tambahan untuk menunjukkan detail dari sebuah bagian proses bisnis.	
3	<i>Group</i>	Merupakan pengelompokan <i>element</i> dari BPMN secara <i>informal</i> .	

Sumber : Object Management Group (2011)

## 2.9 Permodelan *Use Case*

Permodelan *use case* merupakan cara yang mudah untuk mendeskripsikan kebutuhan dari sebuah sistem. Permodelan ini dapat menggambarkan perilaku sistem ketika berinteraksi dengan aktor atau dengan sistem lainnya. Tujuan dari adanya permodelan *use case* adalah agar setiap kebutuhan fungsional dari sebuah sistem yang akan dikembangkan lebih mudah dan tepat (Bittner & Spence, 2002).

Bittner dan Spence juga berpendapat bahwa *use case* sangat berpengaruh karena mampu menunjukkan bagaimana sistem dapat memberikan nilai kepada pemangku kepentingan dan mengetahui siapa saja calon pengguna sistem. Banyak pengembangan perangkat lunak yang mendefinisikan secara detail *use case* tanpa mempertimbangkan pemangku kepentingan, tujuan, kebutuhan nyata, dan permasalahan yang ada. Hal semacam itu dapat berujung pada sebuah sistem yang tidak memenuhi atau sesuai dengan apa yang diharapkan. Sehingga terdapat beberapa tahap yang sebaiknya dilakukan dalam melakukan permodelan *use case*.

### 2.9.1 Identifikasi Tipe Pemangku Kepentingan

Pemangku kepentingan merupakan seseorang yang memperoleh dampak secara materil dari sistem yang akan dikembangkan. Gagasan tersebut mencerminkan fakta bahwa pemangku kepentingan merupakan individu yang secara langsung mendapatkan dampak dari sistem dan secara tidak langsung mempengaruhi sistem yang dikembangkan (Bittner & Spence, 2002). Identifikasi pemangku kepentingan merupakan langkah awal untuk mengetahui siapa saja pemangku kepentingan dari sistem yang akan dikembangkan. Hasil dari

identifikasi tersebut akan dikelompokkan berdasarkan karakteristik dan hubungan kepada sistem.

**Table 2.8 Analisis Tipe Pemangku Kepentingan**

Tipe Pemangku Kepentingan	Deskripsi	Pemangku Kepentingan
[Tipe pemangku kepentingan]	[Menjelaskan tipe pemangku kepentingan]	[Menyebutkan pemangku kepentingan yang termasuk dalam tipe pemangku kepentingan]

Sumber : Diadaptasi dari Bittner & Spence (2003)

### 2.9.2 Analisis Masalah

Analisis masalah merupakan sebuah proses pemahaman mengenai masalah sesungguhnya, kebutuhan pengguna, serta bagaimana kebutuhan tersebut dapat dipenuhi (Leffingwell & Widrig, 2003). Tujuan dari proses ini adalah untuk mengetahui pemahaman yang mendalam terhadap permasalahan sebelum pengembangan sistem dilakukan. Dalam analisis masalah identifikasi aktor merupakan hal yang penting, karena aktor yang terlibat dapat mengerti akar dari permasalahan yang ada. Cara terbaik untuk mendefinisikan analisis permasalahan adalah dengan membuat sebuah *problem statement* yang merupakan ringkasan dari permasalahan, dampak, dan solusi dari permasalahan (Bittner & Spence, 2002).

**Table 2.9 Analisis Masalah**

Masalah	Berisi diskripsi dari permasalahan
Mempengaruhi	Berisi pemangku kepentingan yang terpengaruh dengan adanya permasalahan
Dampak	Berisi dampak masalah terdapat pemangku kepentingan dan proses bisnis
Solusi	Berisi solusi yang ditawarkan untuk menyelesaikan permasalahan

Sumber : Diadaptasi dari Bittner & Spence (2003)

### 2.9.3 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna

Melakukan identifikasi kebutuhan dari pemangku kepentingan dan pengguna merupakan sebuah cara efektif untuk menyelesaikan permasalahan yang kompleks (Bittner & Spence, 2002). Setiap pemangku kepentingan akan memiliki pandangan berbeda mengenai permasalahan dan solusi yang dibutuhkan. Kebutuhan pemangku kepentingan memiliki prioritas untuk memastikan fokus dari pengerjaan proyek hal ini merupakan bagian paling penting dalam sistem, sehingga tim proyek dapat mengetahui bagian yang perlu dikerjakan terlebih dahulu. Salah satu teknik yang digunakan untuk prioritas

kebutuhan adalah aturan *MoSCow*. Aturan *MoSCow* merupakan aturan yang membagi kebutuhan menjadi empat prioritas yaitu (Bittner & Spence, 2002):

- a. *Must Have (Mo)*  
*Must Have* memiliki arti bahwa suatu fitur dalam proyek harus diselesaikan dan dapat bekerja dengan baik ketika proyek selesai. Fitur tersebut penting karena merupakan solusi dari penyelesaian masalah atau fitur utama.
- b. *Should Have (S)*  
*Should Have* memiliki arti bahwa suatu fitur tidak memiliki prioritas utama dalam pengerjaan sistem, tetapi penting dan memiliki nilai lebih bagi pengguna.
- c. *Could Have (Co)*  
*Could Have* memiliki arti bahwa suatu fitur yang dapat meningkatkan pengguna sistem, namun ketika tidak dapat diimplementasikan tidak berpengaruh pada proyek.
- d. *Won't Have This Time (W)*  
*Won't Have This Time* memiliki arti bahwa suatu fitur dapat diimplementasikan setelah pengembangan selesai

**Table 2.10 Identifikasi Kebutuhan Pemangku Kepentingan dan Pengguna**

Kebutuhan	Pemangku Kepentingan	Prioritas	Situasi Saat ini	Solusi yang di tawarkan
Berisi diskripsi dari kebutuhan	Berisi pemangku kepentingan yang berperan	Berisi skala prioritas	Berisi diskripsi dari permasalahan yang ada pada saat ini	Berisi diskripsi dari solusi yang ditawarkan

Sumber : Diadaptasi dari Bittner & Spence (2003)

#### 2.9.4 Identifikasi Fitur

Fitur adalah sebuah kemampuan sistem untuk menyediakan nilai bagi pengguna serta memenuhi kebutuhan pemangku kepentingan dan pengguna. Sebuah set fitur juga dapat menggambarkan ringkasan manfaat yang ditawarkan dari sistem yang dibangun (Bittner & Spence, 2003). Fitur harus dalam lingkup detail yang umum agar semua pemangku kepentingan dapat mengerti namun juga mampu menyediakan informasi bagi tim pengembang untuk membentuk, memvalidasi dan mengelola permodelan *use case*.

**Table 2.11 Fitur Sistem**

Kode Fitur	Fitur	Diskripsi	Prioritas
Kode setiap fitur	Fitur sistem	Diskripsi dari fitur	Prioritas fitur

Sumber : Bittner & Spence (2003)

### 2.9.5 Identifikasi Kebutuhan Fungsional dan Kebutuhan Non Fungsional

Kebutuhan fungsional merupakan layanan yang harus dimiliki sistem, kebutuhan fungsional juga merupakan pernyataan yang menyatakan bagaimana sistem merespon *input* tertentu dan bagaimana sistem tersebut berjalan dalam sebuah kondisi tertentu (Sommerville, 2011). Sebuah kebutuhan fungsional yang baik harus lebih spesifik dalam menggambarkan fungsi – fungsi sistem mencakup *input* dan *output* yang ada pada sistem tersebut.

Kebutuhan non fungsional merupakan sebuah kebutuhan yang tidak secara spesifik disampaikan oleh sistem kepada penggunaannya. Contoh dari sebuah kebutuhan nonfungsional antara lain *performance*, *security* atau *availability* (Sommerville, 2011). Namun kebutuhan non fungsional sangat berpengaruh pada sistem, terkadang jika kebutuhan non fungsional tersebut tidak terpenuhi maka sistem tidak dapat berjalan. Implementasi kebutuhan non fungsional lebih sulit karena bersifat menyeluruh dan terkadang saling berkaitan.

### 2.10 Unified Modeling Language (UML)


*Unified Modeling Language (UML)* merupakan sebuah notasi bagi *developer* atau pengembang untuk menggambarkan skenario dan merumuskan arsitektur sehingga mudah untuk dipahami. UML memiliki sifat yang sederhana namun ekspresif sehingga mempermudah pemeriksaan terdapat konsistensi dan kebenaran dari skenario yang dibuat. Saat ini *UML* merupakan bahasa pemodelan utama untuk menganalisa, menentukan dan merancang sistem perangkat lunak (Booch, 1998).

UML bahasa pemodelan terpadu yang berisikan notasi untuk permodelan dan pengembangan sistem berorientasi objek. UML menyediakan teknologi untuk mendukung pengembangan rekayasa perangkat lunak akan tetapi tidak menyediakan kerangka kerja proses untuk manajemen tim. *Unified Modelling Language (UML)* memiliki berbagai jenis diagram diantaranya *use case diagram*, *activity diagram*, *class diagram*, dan *sequence diagram* (Pressman, 2010).



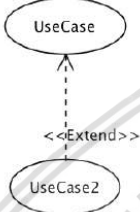
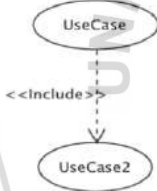

#### 2.10.1 Use Case Diagram

*Use Case Diagram* adalah permodelan dari kelakuan (*behavior*) sistem informasi yang akan dikembangkan. *Use case* mendiskripsikan suatu interaksi antara *actor* dengan sistem. Secara singkat *use case* digunakan untuk mengetahui fungsi – fungsi apa saja yang ada pada sebuah sistem dan siapa saja yang berhak menggunakan fungsi tersebut (Sukanto dan Shalahuddin, 2014).

Table 2.12 Simbol *Use Case Diagram*

No	Simbol	Deskripsi
1		<i>Use case</i> fungsional yang disediakan oleh sistem dengan penamaan menggunakan kalimat kerja.



2	<b>Actor</b> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem yang dikembangkan. Penamaan <i>actor</i> menggunakan kata benda.
3	<b>Association</b> 	Komunikasi antara <i>actor</i> dan <i>use case</i> .
4	<b>Extend</b> 	<i>Extend</i> merupakan sebuah hubungan dengan menambahkan sebuah <i>use case</i> tambahan kepada sebuah <i>use case</i> . Dimana <i>use case</i> utama dapat berdiri sendiri tanpa <i>use case</i> tambahan dilakukan.
5	<b>Include</b> 	<i>Include</i> merupakan sebuah hubungan dengan menambahkan sebuah <i>use case</i> tambahan yang wajib untuk dijalankan ketika <i>use case</i> utama dilakukan.
6	<b>Generalisasi</b> 	Merupakan notasi hubungan antar aktor yang menghubungkan aktor yang lebih spesifik dengan aktor yang umum.

Sumber: Diadaptasi dari Object Management Group (2005)

Pada *use case diagram* tidak menyediakan detail informasi mengenai sebuah *use case* tertentu. Sehingga dalam sebuah *use case diagram* harus didukung dengan sebuah permodelan yang memiliki spesifikasi *use case*, dimana berisi penjelasan bagaimana aktor dan sistem berkolaborasi untuk mencapai tujuan yang diharapkan dari tiap *use case* tersebut.

**Table 2.13 Skenario Use Case**

<i>Use Case Name</i>	
<i>Brief Description</i>	Berisi penjelasan singkat dari sebuah <i>use case</i> .
<i>Actor</i>	Berisi aktor yang terlibat dalam <i>use case</i> .
<i>Pre-conditions</i>	Berisi satu atau lebih kondisi yang harus dipenuhi sebelum <i>use case</i> dapat dijalankan.








<i>Basic flow of Events</i>	Berisi alur normal yang dilakukan oleh <i>use case</i> .
<i>Alternative Flows</i>	Berisi alur opsional ketika sistem tidak berjalan sesuai dengan alur semestinya.
<i>Post-conditions</i>	Berisi satu atau lebih keadan yang terjadi ketika <i>use case</i> telah berakhir

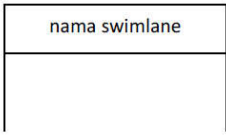
Sumber: Diadaptasi dari Bittner & Spence (2002)

### 2.10.2 Activity Diagram

*Activity diagram* menggambarkan sebuah aliran kerja atau *workflow* dari sebuah sistem. Sehingga *activity diagram* dapat menggambarkan aliran kerja dari sebuah aktifitas di dalam sistem dan siapa yang melakukan hal tersebut (Sukanto dan Shalahuddin, 2014).

**Table 2.14 Simbol Activity Diagram**

No	Simbol	Deskripsi
1	Status awal 	Status awal dari sebuah aktivitas sistem.
2	Aktivitas 	Aktivitas yang dilakukan didalam sistem yang biasanya diawali dengan kata kerja.
3	Percabangan/ <i>Decision</i> 	<i>Asosiasi</i> dimana jika terdapat percabangan pilihan 2 atau lebih aktivitas.
4	Penggabungan / Join 	<i>Asosiasi</i> penggabungan dari hasil perpecahan percabangan.
5	Status akhir 	Sebuah akhir atau hasil dari sebuah aktifitas dalam sebuah sistem.




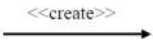
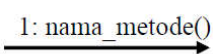
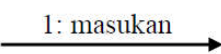
6	<b>Swimlane</b> 	Pemisah atau pembagi organisasi bisnis yang bertanggung jawab kepada aktifitas yang ada.
---	--	--

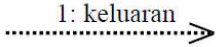
Sumber: Sukamto dan Shalahuddin (2014)

### 2.10.3 Sequence Diagram

*Sequence diagram* adalah sebuah notasi untuk menggambarkan interaksi objek pada *use case* dengan mendeskripsikan waktu hidup serta pesan yang dikirim antar objek. Untuk membuat *sequence diagram* hal terpenting adalah mengetahui objek – objek apa yang terlibat beserta metode – metode yang dimiliki (Sukamto dan Shalahuddin, 2014).

Table 2.15 Simbol dalam *Sequence Diagram*

No	Simbol	Deskripsi
1	Aktor 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem dikembangkan
2	Objek 	Menyatakan objek yang berinteraksi.
3	Waktu aktif 	Menyatakan sebuah objek dalam keadaan aktif dan berinteraksi
4	Pesan tipe <i>create</i> 	Menyatakan suatu objek membuat objek lain, digambarkan dengan arah panah mengarah pada objek yang dibuat.
5	Pesan tipe <i>call</i> 	Menyatakan suatu objek memanggil operasi atau metode yang ada pada objek lain atau dirinya sendiri.
6	Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirimkan <i>input</i> ke objek lainnya, digambarkan dengan arah panah mengarah pada objek yang menerima pesan.

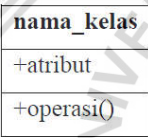
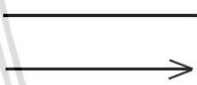


7	Pesan tipe <i>return</i> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ( <i>return</i> ).
---	---	---

Sumber: Sukamto & Shalahudin (2014)

#### 2.10.4 Class Diagram

*Class diagram* berfungsi untuk menunjukkan keberadaan dan hubungan setiap *class* yang membentuk arsitektur sistem (Booch, et al., 1998). Terdapat dua elemen utama dalam pembentukan *class diagram* yaitu *class* dan *relationships*. Sukamto dan Shalahudin (2014) mengartikan *class diagram* sebagai notasi untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

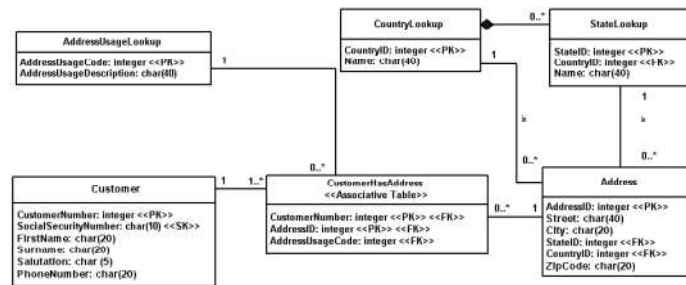
Table 2.16 Simbol *Class Diagram*

No	Simbol	Deskripsi
1	Kelas 	Kelas pada struktur sistem. Terdiri dari nama kelas, atribut dan operasi yang ada pada kelas tersebut.
3	Asosiasi / Association 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5	Generalisasi 	Relasi antarkelas dengan makna generalisasi (umum khusus).
6	Aggregation 	Relasi antarkelas dengan makna semua-bagian ( <i>whole-part</i> ).

Sumber: Sukamto & Shalahudin (2014)

#### 2.11 Physical Data Model

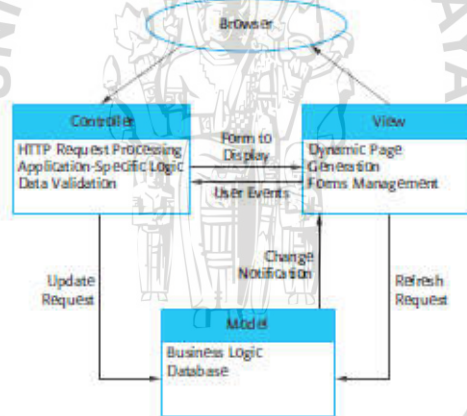
*Physical data model* (PDM) digunakan untuk merancang skema internal *database*. PDM menggambarkan tabel data, kolom data, dan hubungan antar tabel (www.agiledata.org). Dalam sebuah pengembangan sistem PDM sangat berguna karena merupakan acuan dalam pembuatan data base dari sistem tersebut.

Gambar 2.2 Contoh *Physical Data Model*

Sumber: www.agiledata.org

## 2.12 Model-View-Controller

*Model-View-Controller* (MVC) merupakan salah satu pola arsitektur dimana membagi sebuah sistem menjadi 3 komponen logis yang saling berkaitan (Sommerville, 2011). Salah satu kelebihan dari pola MVC adalah independen sehingga mempermudah *custom* pada data yang ditampilkan. Dalam bukunya Sommerville menjelaskan bahwa ketiga komponen tersebut mempunyai fungsi yang berberda-beda.

Gambar 2.3 Skema *Model View Controller*

Sumber: Sommerville (2011)

Tiga komponen utama dalam MVC adalah *Model*, *View* dan *Controller*. Komponen *model* berfungsi untuk mengelola data pada sistem dan operasi terkait data tersebut. Didalam *model* terdapat logika bisnis atau fungsi untuk mengelola *data base*. Pada komponen *view* peran utamanya adalah menampilkan informasi kepada pengguna sistem. Setiap informasi yang ditampilkan pada komponen *view* dipengaruhi oleh data yang diperoleh dari komponen *model*. Sedangkan komponen *controller* memiliki peran untuk mengelola interaksi pengguna pada komponen *view*. Secara tidak langsung komponen *controller* merupakan jembatan antara presentasi dan data base.

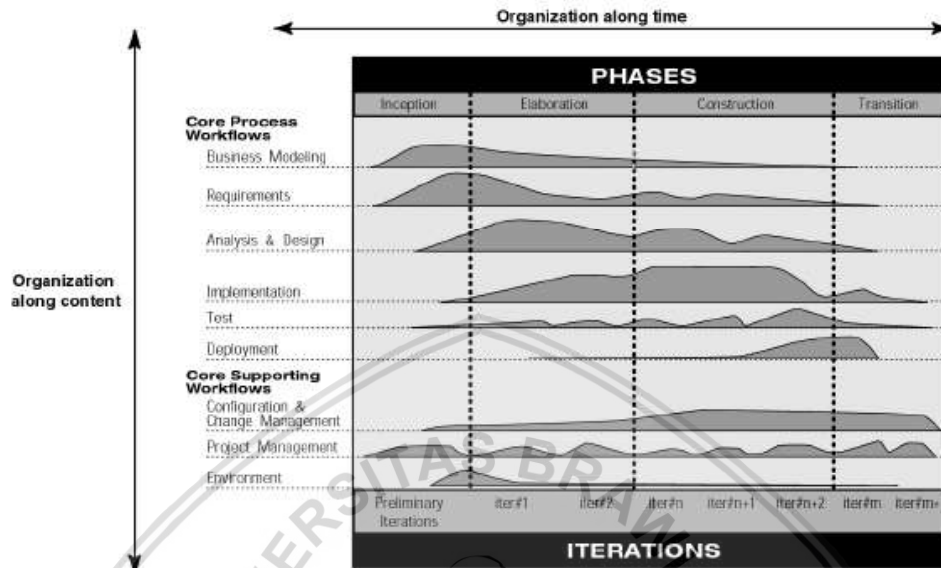
### 2.13 Rational Unified Process (RUP)

*Rational Unified Process* adalah sebuah metode dalam pengembangan perangkat lunak dengan menggunakan pendekatan *iterative* (perulangan). Metode ini mengusung pengembangan perangkat lunak dengan membagi menjadi beberapa fase. RUP menerapkan konsep *object-oriented* yang pada pengembangannya berfokus pada *Unified Model Language (UML)* (Jalote, 2008). *Iterative* dalam metode RUP merupakan pembeda dengan metode pengembangan lainnya, berbeda dengan *waterfall* dimana iteratif bersifat keseluruhan metode RUP mengakomodasi iteratif pada tiap fasenya. Selain itu pada RUP memisahkan tahapan dan aktifitas, dimana pada tiap fase terdapat aktifitas utama dan sub aktifitas dari fase lainnya.

Tujuan dari *Rational Unified Process (RUP)* adalah menciptakan sebuah perangkat lunak berkualitas tinggi yang memenuhi kebutuhan *end user* dalam jadwal dan anggaran yang dapat diprediksi. RUP yang bersifat *iterative* dibagi menjadi beberapa fase – fase yang mempunyai fokus tersendiri menekankan pada pembangunan sistem berdasarkan pemahaman secara keseluruhan tentang bagaimana sistem tersebut akan digunakan. Penggunaan *use case* dan *scenario* digunakan untuk menyelaraskan antara alur suatu proses dari kebutuhan yang didapat. Dalam segi pengembangan yang berfokus pada kebutuhan pengguna maka RUP merupakan salah satu metode pengembangan terbaik saat ini (Booch et al, 1998). Metode RUP memiliki karakteristik yang dapat membedakan dengan metode pengembangan lainnya antara lain :

1. *Architecture-Centric* yang berarti arsitektur sistem dijadikan sebagai patokan penentuan konsep, konstruksi, pengelolaan dan perbaikan sistem selama pengembangan berlangsung.
2. *Use Case driven* yang berarti penggunaan *use case* sebagai patokan terhadap perilaku sistem yang akan dikembangkan.
3. *Object-Oriented Techniques* yang berarti pengembangan berdasar pada konsep objek dan kelas yang saling berhubungan dan menggunakan UML sebagai notasi umumnya.
4. *Configurable Process* yang berarti metode ini dapat disesuaikan berdasarkan proyek yang dikembangkan.
5. *Quality Control And Risk Management* yang berarti metode ini dapat memberikan kontrol terhadap sistem yang dikembangkan dan resiko yang ada karena pada metode RUP semua kegiatan melibatkan pemangku kepentingan.

### 2.13.1 Fase – Fase pada Metode RUP



Gambar 2.4 Fase – Fase pada Metode RUP

Sumber : IBM (1998)

Fase – fase pada *Rational Unified Process (RUP)* digambarkan pada sumbu horizontal yang berkaitan dengan waktu. Pembagian tiap fase berdasarkan pada fokus aktifitas utama yang dikerjakan pada tiap fasenya. Akan tetapi RUP dapat mengakomodasi sub aktifitas seperti yang digambarkan pada gambar 2.4. Berikut merupakan fase – fase yang ada pada metode *Rational Unified Process (RUP)* (IBM,1998) :

#### 1. Insepsi

Tahapan ini merupakan tahapan dimana terdapat penetapan permasalahan, *business case*, dan batasan ruang lingkup proyek. Untuk dapat melakukan identifikasi terhadap semua permasalahan diatas perlu dilakukan analisi terhadap seluruh aspek yang berkaitan dengan sistem yang akan dikembangkan. Hasil dari fase *inception* adalah :

- Gambaran umum mengenai sistem meliputi kebutuhan sistem, fitur utama dan kendala utama.
- Use case telah dibuat (10% - 20%).
- Mendefinisikan kemungkinan masalah
- Mendefinisikan resiko.
- Melakukan perancangan proyek meliputi fase dan literasi yang terjadi.



- Pemodelan bisnis.

## 2. Elaborasi

Tahap ini lebih fokus kepada analisis dan perancangan sistem yang akan dikembangkan berdasarkan permodelan *use case* pada tahap sebelumnya. Fase elaborasi mengakomodasi perubahan kebutuhan dari fase sebelumnya berdasarkan perubahan kebutuhan dari pemangku kepentingan. Dalam fase ini harus memiliki satu atau dua iterasi, sehingga pada akhir fase sudah didapatkan sebuah kebutuhan dan rancangan sistem yang stabil. Hasil dari fase elaborasi antara lain :

- Pemodelan *use case* telah mencapai 80% selesai.
- Mendefinisikan arsitektur sistem.
- Mendefinisikan rancangan pengembangan sistem secara menyeluruh.

## 3. Konstruksi

Tahapan ini adalah tahapan dilakukannya pembuatan sistem secara keseluruhan. Fokus utamanya adalah pengembangan dan perbaikan rancangan sistem berdasar pada rancangan yang didefinisikan pada fase sebelumnya. Pada fase ini permodelan bisnis dan kebutuhan harus dalam kondisi stabil. Aktivitas pengujian juga terfokus pada fase ini. Hasil dari fase konstruksi antara lain:

- Produk perangkat lunak harus sudah dapat digunakan.
- Panduan pengguna.
- Deskripsi versi dari produk perangkat lunak.

## 4. Transisi

Pada tahapan akhir dalam metode RUP ini sistem yang dikembangkan dapat diberikan kepada pengguna akhir. Hal tersebut dapat memberikan *feedback* sehingga dapat mengetahui kekurangan yang nantinya akan diselesaikan pada versi berikutnya. Dapat diartikan pada tahapan ini mengharuskan sistem telah selesai dan mendapatkan tingkat penerimaan tinggi dari pengguna. Hasil dari fase transisi adalah :

- *Beta testing* untuk mengetahui validasi sistem terhadap penggunaanya.
- Peralihan dari sistem sebelumnya pada sistem yang baru.
- Penerapan data sesungguhnya.
- Pelatihan dan pengelolaan sistem.

### 2.13.2 Aktivitas dalam Metode RUP

Terdapat sembilan aktifitas yang ada pada metode *Rational Unified Process (RUP)*. Aktifitas - aktifitas tersebut dikelompokkan menjadi 2 kategori utama yang berdasarkan dari jenis aktifitasnya. Sehingga terdapat 6 aktifitas yang tergabung pada *core process workflow* dimana didalamnya merupakan aktifitas yang berkaitan dengan pengembangan sistem. Sedangkan 3 aktifitas lainnya tergolong dalam *core supporting workflow* yang lebih berkaitan dengan manajemen dari proyek tersebut. Berikut merupakan aktifitas – aktifitas yang terdapat pada metode *Rational Unified Process (IBM,1998)*:

1. Pemodelan bisnis.

Melakukan pemodelan bisnis bertujuan untuk memahami lebih dalam mengenai proses bisnis yang berkaitan dengan sistem yang akan dibuat. Sehingga terdapat pemahaman yang sama pada semua pemangku kepentingan. Namun tidak semua proyek menerapkan aktifitas ini.

2. Kebutuhan

Aktifitas ini bertujuan untuk menggambarkan apa saja yang dapat dilakukan oleh sistem. Tahapan ini terdapat identifikasi terhadap aktor dan *use case*. Sehingga dapat diketahui interaksi antara aktor dan sistem.

3. Analisis dan Perancangan

Tujuan aktifitas ini adalah untuk menggambarkan bagaimana sistem akan dibuat pada tahap implementasi. Perancang terdiri dari desain struktur *class* dan tampilan yang nantinya akan diterapkan. Perancangan berdasar pada arsitektur utama dan desain perancangan merupakan fokus utama iterasi.

4. Implementasi

Implementasi dilakukan untuk merealisasikan perancangan sistem yang telah dibuat menjadi sebuah sistem nyata.

5. Pengujian

Pengujian bertujuan untuk melakukan validasi terhadap integrasi yang ada pada sistem. Selain itu juga dilakukan validasi terhadap semua kebutuhan yang telah di definisikan pada awal pengembangan sistem. Secara garis besar pengujian dilakukan terhadap 3 ukuran kualitas *functionality*, *application performance* dan *system performance*.

6. *Deployment*

*Deployment* merupakan aktifitas untuk merilis produk yang telah dibangun sebelumnya.

7. Konfigurasi dan manajemen perubahan

Aktifitas ini bertujuan untuk mengontrol berbagai artefak yang didefinisikan oleh tim pengembang secara keseluruhan.

## 8. Manajemen Proyek

Aktivitas ini merupakan aktivitas pendukung mengenai penyelarasan antara tujuan utama, manajemen resiko, dan manajemen terhadap kendala yang ada.

## 9. *Environment*

Aktivitas ini merupakan aktivitas pendukung yang berkaitan dengan konfigurasi proses dalam proyek selain itu juga berfokus untuk membuat pedoman *guideline* dalam proyek.

## 2.14 Pengujian

Pengujian perangkat lunak merupakan sebuah tahapan untuk bahwa perangkat lunak yang telah dikembangkan dapat berjalan sesuai harapan dan untuk mengetahui cacat (*defects*) (Sommerville, 2011). Dalam pengujian perangkat lunak terdapat berbagai jenis pengujian dengan fokus yang berbeda – beda. Secara garis besar dibagi menjadi 2 jenis yaitu *development testing* dan *user testing*.

### 2.14.1 *Black Box Testing*

Pengujian *black box* juga dikenal sebagai *behavioral testing* merupakan pengujian perangkat lunak yang berfokus pada persyaratan fungsional. Pengujian *black box* dilakukan dengan cara menentukan set *input* yang nantinya akan diproses oleh sistem dengan sepenuhnya menjalankan semua persyaratan fungsional. Pengujian ini tidak memperdulikan struktur dari sistem, melainkan berfokus pada domain informasi (Pressman, 2010).

Pengujian *black box* berusaha untuk menemukan kesalahan dalam beberapa katagori sebagai berikut :

1. Fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses database eksternal.
4. Kesalahan performa.
5. Kesalahan inisialisasi dan kesalahan terminasi.

### 2.14.2 *Compatibility Testing*

*Compatibility tesing* merupakan pengujian sebuah *web* dengan menjalankan pada berbagai *host* yang berbeda. Tujuannya adalah untuk menemukan kesalahan yang khusus pada konfigurasi *host* (Pressman, 2010). Pengujian ini pada dasarnya dilakukan untuk mengetahui performa dari sistem pada sisi *client* atau pengguna yang memiliki berbagai jenis cara untuk mengakses sistem tersebut.

Konfigurasi dari pengujian ini dapat meliputi penggunaan sistem operasi, jenis perangkat, aplikasi peramba atau kecepatan yang berbeda. Salah satu

*software* yang dapat membantu dalam *compatibility testing* adalah *sortsite*. *Software* tersebut dapat melakukan serangkaian pengujian dengan cara memverituaikan berbagai konfigurasi.

### 2.14.3 User Acceptance Testing

*User Acceptance Testing* (UAT) merupakan pengujian tingkat penerimaan sistem berdasarkan harapan *user* dari sistem tersebut. Pemangku kepentingan harus memastikan bahwa sistem yang telah dibangun dapat memenuhi kebutuhan dan harapan dari pengguna akhir agar sistem tersebut diterima atau digunakan. Pengujian UAT harus dapat diukur sehingga dapat diketahui sejauh mana sistem diterima oleh *end user* (Kshirasagar, 2008).

Pada pengujian *User Acceptance Testing* terdapat beberapa kriteria yang membantu dalam menilai penerimaan sistem. Namun dalam penerapannya tidak semua kriteria harus diuji pada sistem karena akan mengakibatkan terlalu banyak dan umumnya hasil pengujian. Tim pengembang dapat memilih bagian – bagian penting yang dapat mewakili penerimaan sistem. Berikut merupakan kriteria – kriteria yang ada pada pengujian UAT.

**Table 2.17 Kriteria User Acceptance Testing**

Kriteria	Diskripsi
<i>Functional Correctness and Completeness</i>	Berfokus pada apakah sistem tersebut telah sesuai harapan dengan mengacu pada fitur yang telah didefinisikan.
<i>Accuracy</i>	Berfokus apakah perhitungan atau hasil dari sistem sesuai dengan nilai seharusnya.
<i>Data Integrity</i>	Berfokus pada performa sistem dalam mengolah data.
<i>Data Conversion</i>	Berfokus pada performa sistem dalam menjaga data pada saat terjadi perubahan jenis atau bentuk.
<i>Backup and Recovery</i>	Berfokus pada pengcadangan dan pemulihan data
<i>Competitive Edge</i>	Berfokus pada keunggulan dibandingkan model yang lain.
<i>Usability</i>	Berfokus pada kemudahan dalam penggunaan sistem.
<i>Performance</i>	Berfokus pada kinerja sistem.

Sumber: Diadaptasi dari Kshirasagar (2008)

Dalam proses *User Acceptance Testing* terdapat beberapa tahap yang harus dilakukan (Sommerville, 2011):

1. *Define acceptance criteria*

Tahapan awal dalam UAT adalah mendefinisikan kriteria apa yang akan dijadikan pedoman dalam pengujian.

2. *Plan acceptance testing*

Tahap perencanaan mengenai biaya, waktu dan partisipan dalam pengujian yang akan dilakukan.

3. *Derive acceptance tests*

Pada tahap ini terdapat perencanaan lebih mendetail mengenai pengujian sehingga pada akhirnya dapat mewakili pengujian pada seluruh kebutuhan fungsional dan non fungsional.

4. *Run acceptance tests*

Tahapan dimana pengujian dilakukan dan melibatkan calon pengguna. Idealnya sistem dijalankan dalam lingkungan sebenarnya namun hal tersebut mungkin dapat mengganggu atau membutuhkan sumber daya yang besar.

5. *Negotiate test results*

Tahapan dimana hasil dari pengujian diolah sedemikian rupa sehingga dapat diambil sebuah nilai besar penerimaan terhadap sistem.

6. *Reject/accept system*

Merupakan tahap akhir dimana pemangku kepentingan akan memberikan keputusan apakah sistem dapat dijalankan atau harus dilakukan perbaikan.

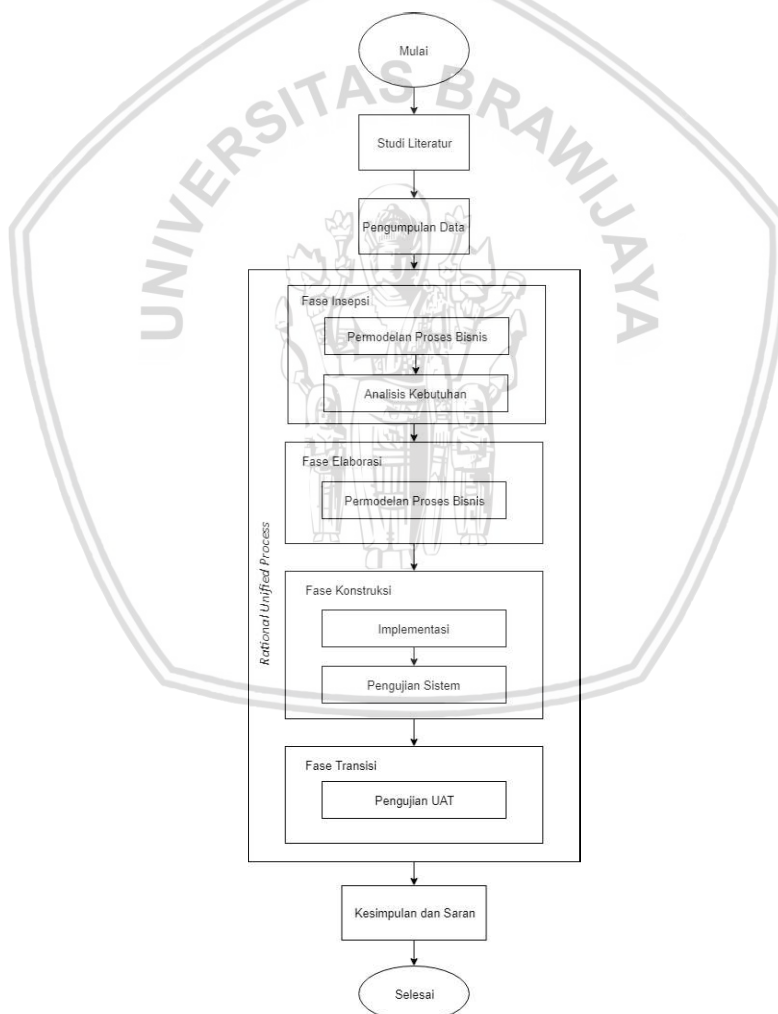
## BAB 3 METODOLOGI

### 3.1 Tempat dan Waktu Penelitian

Penelitian ini dilakukan pada bulan September 2017 hingga bulan Maret 2018. Dalam melakukan studi lapangan di KPRI “GURU” Kecamatan Sumbermanjing Wetan, Malang, Jawa Timur.

### 3.2 Langkah – Langkah Penelitian

Penelitian pembangunan sistem informasi *customer-touching applications* ini merupakan tipe penelitian implementatif. Penulis melakukan penelitian pada KPRI “GURU” Kecamatan Sumbermanjing Wetan. Langkah – langkah dalam melakukan penelitian adalah:



Gambar 3.1 Diagram Penelitian



### 3.3 Studi Literatur

Tahap pertama yang dilakukan dalam penelitian adalah studi literatur. Studi Literatur dalam penelitian ini bertujuan untuk mendapatkan gambaran yang menyeluruh tentang apa yang sudah dikerjakan oleh orang lain dan bagaimana cara pengerjaannya. Studi literatur dilakukan untuk mendapatkan landasan kerangka berpikir atau metode untuk mendukung penelitian. Studi literatur diperoleh dari buku, laporan penelitian, jurnal, *internet* dan berbagai sumber lainnya. Tujuan dari tahap ini adalah memperoleh informasi yang menunjang proses penelitian.

### 3.4 Pengumpulan Data

Tahap pengumpulan data termasuk dalam tahap yang penting dimana peneliti akan mengumpulkan informasi proses bisnis dan informasi yang dibutuhkan untuk analisa sistem yang dikembangkan. Data yang diperoleh dari tahap ini juga berpengaruh terhadap hasil dari penelitian. Metode yang digunakan untuk melakukan pengumpulan data pada penelitian ini antara lain:

#### 3.4.1 Wawancara

Wawancara dilakukan terhadap pihak yang terkait dengan sistem yang saat ini ada dan pihak yang nantinya akan terkait dengan sistem. Wawancara berfungsi untuk mendapatkan data yang nantinya akan dijadikan dasar dalam pembangunan sistem. Data yang didapatkan dalam wawancara berupa gambaran kebutuhan dan gambaran proses bisnis. Wawancara dilakukan dengan menayakan proses bisnis, permasalahan yang ada, dan kondisi dari KPRI untuk menerapkan sistem yang akan dibangun. Wawancara akan dilakukan kepada pengawas unit simpan pinjam sehingga dapat mengetahui proses bisnis yang ada dan kebutuhan fungsional maupun non fungsional yang dibutuhkan.

#### 3.4.2 Observasi

Observasi dilakukan untuk mengetahui proses bisnis yang ada. Dengan menggunakan metode ini peneliti dapat mendapatkan data dari memperhatikan proses bisnis yang terjadi sehingga dapat mengetahui masalah yang benar – benar terjadi. Selain itu observasi yang dilakukan penulis bertujuan untuk mengumpulkan data dengan lingkungan dan aktivitas kerja dari bahan kajian.

### 3.5 Fase Insepsi

Fase insepsi merupakan fase pertama dalam metode *Relational Unified Process* (RUP). Pada penelitian ini fase insepsi ini berfokus pada *Bussines Modeling* dan *Requirments* dari sistem yang akan dikembangkan. Pada fase tidak terdapat *literasi* didalamnya. *Bussines Modeling* diperlukan untuk memastikan terdapat pemahaman yang sama tentang apa saja yang harus diperlukan untuk mendukung proses bisnis sedangkan *requirments* mendiskripsikan kebutuhan dari aplikasi yang akan dikembangkan pada Unit Simpan Pinjam KPRI.

### 3.5.1 Permodelan Proses Bisnis

Pada tahap ini merupakan proses bisnis untuk pengembangan sistem informasi dari unit simpan pinjam. Analisis *business proses as-is* digunakan untuk menggambarkan proses bisnis yang saat ini pada Unit Simpan Pinjam ada sehingga dapat diketahui apa saja yang harus dilakukan oleh sistem untuk memperbaiki proses bisnis yang ada. Dari hasil analisis *business proses as-is* dibuat *business proses to-be* atau usulan bisnis proses yang nantinya akan diterapkan ketika sistem yang dikembangkan telah diimplementasikan kedalam organisasi.

### 3.5.2 Analisi Kebutuhan

Peneliti akan melakukan identifikasi terhadap aktor yang berperan dalam sistem. Didalam analisa tersebut juga mendeskripsikan kebutuhan sehingga dapat diketahui persyaratan kebutuhan fungsional maupun kebutuhan non fungsional. Pada tahap ini juga terdapat permodelan *use case* yang merupakan dasar dari dasar pengembangan *Relational Unified Process*. Hasil dari tahap ini adalah terdapat pemahaman mengenai apa yang difokuskan dalam sistem yang dikembangkan.

### 3.6 Fase Elaborasi

Merupakan fase kedua pada tahapan metode *Relational Unified Process* (RUP). Pada fase elaborasi peneliti memfokuskan perancangan sistem berdasarkan *use case* yang telah dibuat pada fase sebelumnya. Selain itu pada fase elaborasi juga masih mengakomodasi perbaikan dari perubahan yang mungkin saja muncul dari hasil fase sebelumnya yaitu fase insepisi. Fase elaborasi setidaknya terdapat sebuah literasi di dalamnya.

#### 3.6.1 Perancangan Sistem

Pada tahapan ini peneliti menggambarkan rancangan sistem agar implementasi yang dibuat sesuai dengan kebutuhan sistem pada fase sebelumnya. Perancangan dibuat berdasarkan *use case* yang telah disepakati. Perancangan yang dibuat meliputi perancangan *sequence diagram*, *class diagram*, basis data, algoritma dan perancangan antarmuka. Perancangan bertujuan untuk memberikan gambaran kepada pemangku kepentingan yaitu unit simpan pinjam, bagaimana kebutuhan yang telah didefinisikan akan diterapkan kedalam sistem yang dikembangkan.

### 3.7 Fase Konstruksi

Merupakan fase ketiga dalam metode *Relational Unified Process* (RUP). Fase peneliti berfokus kepada implementasi dan *testing* sistem yang dikembangkan. Pada tahap ini *bussines modeling* dan *requirments* telah stabil. Sedangkan untuk analisis dan perancangan masih memungkinkan terjadinya perubahan sesuai dengan masukan dari pemangku kepentingan. Selain implementasi pada fase ini juga terdapat testing sistem berupa *blackbox* dan *compatibiliy testing*.

### 3.7.1 Implementasi

Merupakan tahapan dimana hasil dari tahap tahap sebelumnya diubah menjadi sebuah sistem yang terbuat dari serangkaian kode pemrograman. Implementasi sistem menggunakan bahasa pemrograman HTML, CSS, *Javascript*. Pada implementasi harus mengacu pada *sequence*, *class diagram*, data model yang ada pada fase elaborasi.

### 3.7.2 Pengujian Sistem

Tahapan pengujian merupakan tahap untuk memastikan bahwa sistem informasi yang dibuat telah sesuai dengan perancangan. Pengujian juga bertujuan untuk mengetahui kesalahan – kesalahan yang ada sehingga ketika sistem digunakan tidak terdapat kesalahan pengolahan data atau informasi. Metode untuk melakukan pengujian menggunakan pengujian *black box* yang bertujuan untuk memeriksa setiap fitur – fitur yang ada dapat berjalan dengan baik. Pengujian *black box* dilakukan oleh peneliti sendiri. Selain itu juga terdapat pengujian *compatibility* yang bertujuan untuk mengetahui bahwa kebutuhan non fungsional dimana sistem dapat dijalankan pada berbagai jenis perangkat.

### 3.8 Fase Transisi

Merupakan fase transisi merupakan tahapan terakhir dalam *Rational Unified Process*. Pada penelitian ini fase transisi peneliti berfokus pada salah satu aktivitas yaitu *beta testing*. Pengujian dilakukan menggunakan metode *User Acceptance Test (UAT)* sebagai penilaian penerimaan sistem oleh pengguna.

#### 3.8.1 Pengujian UAT

*User Acceptance Test (UAT)* dilakukan dengan mengkombinasikan bersama metode kuisisioner. Pengujian dilakukan pada akhir pengembangan sistem dan dilakukan di kantor pusat KPRI “GURU” Kec. Sumbermanjing wetan terhadap 20 koresponden yang telah mencoba sistem berdasarkan test uji yang telah didefinisikan. Koresponden yang telah mencoba sistem akan dibagikan kuisisioner yang berdasarkan pada kriteria uji pada pengujian UAT dan menggunakan skala *guttman* untuk mempermudah penarikan hasil atau kesimpulan.

### 3.9 Kesimpulan dan Saran

Proses pengambilan kesimpulan dan saran merupakan tahapan akhir dari penelitian. Pengambilan kesimpulan berdasarkan pada pengujian yang telah diterapkan sebelumnya. Penulisan saran bertujuan untuk memperbaiki kekurangan yang ada pada penelitian sehingga dapat disempurnakan bagi penelitian selanjutnya.

## BAB 4 ANALISIS

Bab ini membahas analisis proses bisnis dan analisis kebutuhan yang merupakan bagian dari fase inepsi dalam metode *Rational Unified Process (RUP)*. Hasil analisis proses bisnis meliputi analisis proses bisnis *as-is* yang dibandingkan dengan proses bisnis *to-be*. Analisis kebutuhan menghasilkan identifikasi pemangku kepentingan, identifikasi pengguna, identifikasi kebutuhan pengguna, identifikasi fitur, spesifikasi kebutuhan, permodelan *use case* dan permodelan *activity diagram*.

### 4.1 Analisis Proses Bisnis

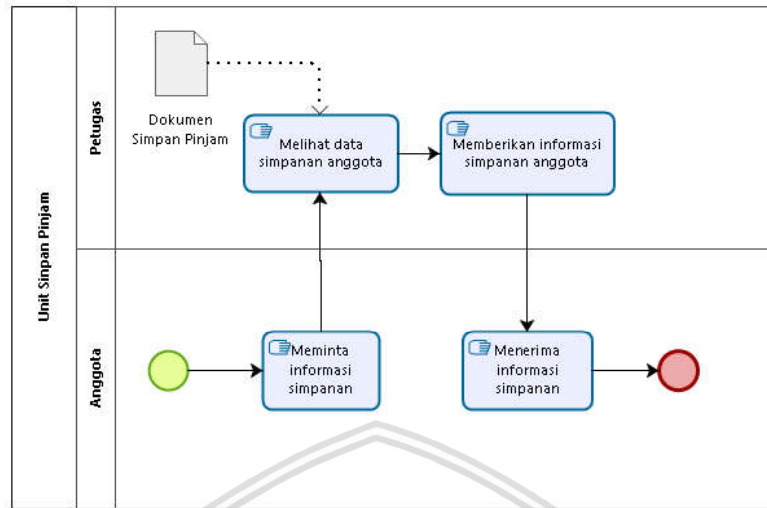
Analisis proses bisnis merupakan tahapan identifikasi terhadap permasalahan proses bisnis yang saat ini dimiliki oleh Unit Simpan Pinjam KPRI. Proses bisnis berdasar pada hasil wawancara dan observasi terhadap Unit Simpan Pinjam. Proses bisnis akan digambarkan menjadi 2 permodelan yaitu permodelan proses bisnis *as-is* (proses bisnis saat ini) dan permodelan bisnis *to-be* (rekomendasi perbaikan proses bisnis).

#### 4.1.1 Pemodelan Proses Bisnis As-Is

Permodelan proses bisnis *as-is* berdasarkan pada keadaan proses bisnis yang saat ini berjalan menggunakan informasi yang didapatkan dari wawancara dan observasi. Pemodelan proses bisnis menggunakan notasi BPMN sehingga lebih mudah dipahami. Proses bisnis *as-is* yang telah dibuat akan divalidasi untuk memastikan bahwa notasi yang dibuat telah sesuai dengan proses yang berjalan. Analisis dan pemodelan proses bisnis *as-is* dilakukan untuk mengidentifikasi permasalahan yang ada pada proses bisnis yang sedang berjalan.

1. Proses bisnis *As-Is* mendapatkan informasi simpanan

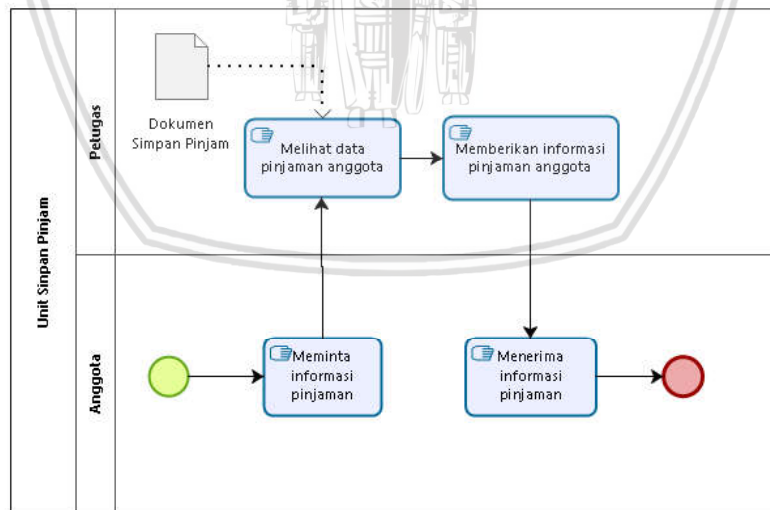
Proses bisnis dimulai dengan anggota datang ke kantor KPRI dan menemui petugas Unit Simpan Pinjam untuk meminta informasi. Anggota akan meminta informasi simpanan kepada petugas. Petugas akan melihat dokumen yang berisi data simpan pinjam untuk mendapatkan informasi yang dibutuhkan kepada anggota. Anggota akan mendapatkan informasi mendetail tentang simpanan yang dimilikinya seperti yang digambarkan gambar 4.1.



**Gambar 4.1 As-is mendapatkan informasi simpanan**

2. Proses bisnis As-is mendapatkan informasi pinjaman

Proses bisnis untuk mendapatkan informasi pinjaman digambarkan pada gambar 4.2. Proses bisnis awali dengan anggota meminta informasi kepada petugas unit simpan pinjam. Petugas akan melihat informasi pinjaman pada dokumen simpan pinjam. Petugas akan memberikan informasi detail mengenai pinjaman yang dimiliki oleh anggota.



**Gambar 4.2 As-is mendapatkan informasi pinjaman**

3. Proses bisnis As-is pengajuan pinjaman

Proses pengajuan pinjaman pada gambar 4.3 dimulai dengan anggota datang ke kantor KPRI dan menemui petugas simpan pinjam untuk



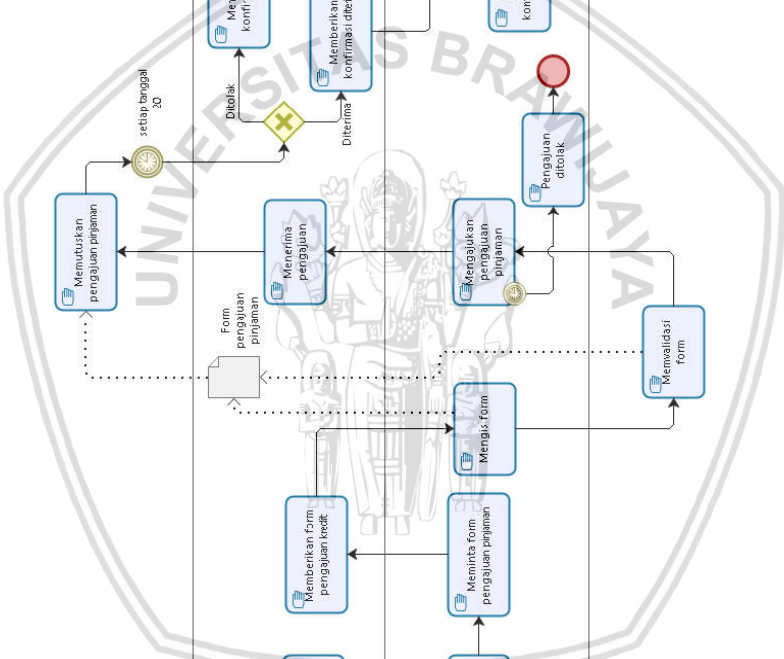
meminta estimasi pinjaman berupa banyak pinjaman yang dapat diajukan, besaran potongan, lama masa pinjaman dan jumlah bunga yang dibebankan. Petugas melihat dokumen simpan pinjam untuk mengetahui berapa simpanan dan pinjaman yang dimiliki oleh anggota. Petugas akan melakukan perhitungan estimasi pinjaman yang dapat diberikan berdasarkan data simpan pinjam. Petugas memberikan form untuk pengajuan pinjaman kepada anggota. Anggota akan mengisi form pengajuan pinjaman sesuai dengan estimasi pinjaman yang dapat diperoleh. Anggota memberikan form untuk divalidasi oleh ketua kelompok dimana validasi bertujuan agar ketua kelompok mengetahui akan adanya pertambahan setoran tiap bulannya.

Setelah form divalidasi anggota harus kembali ke kantor KPRI untuk menyerahkan pengajuan pinjaman. Pengajuan pinjaman akan diputuskan oleh pengurus dan akan diketahui hasilnya pada tanggal 20 tiap bulannya. Hasil putusan dari pengurus akan disampaikan oleh petugas ketika anggota datang untuk menayakan hasil putusan. Ketika pengajuan diterima maka anggota dapat mengajukan pencairan pinjaman. Petugas akan melakukan pencatatan pinjaman baru dan melakukan proses pencairan pinjaman. Setelah proses pencairan pinjaman selesai maka anggota dapat mendapatkan pinjaman yang diajukan.

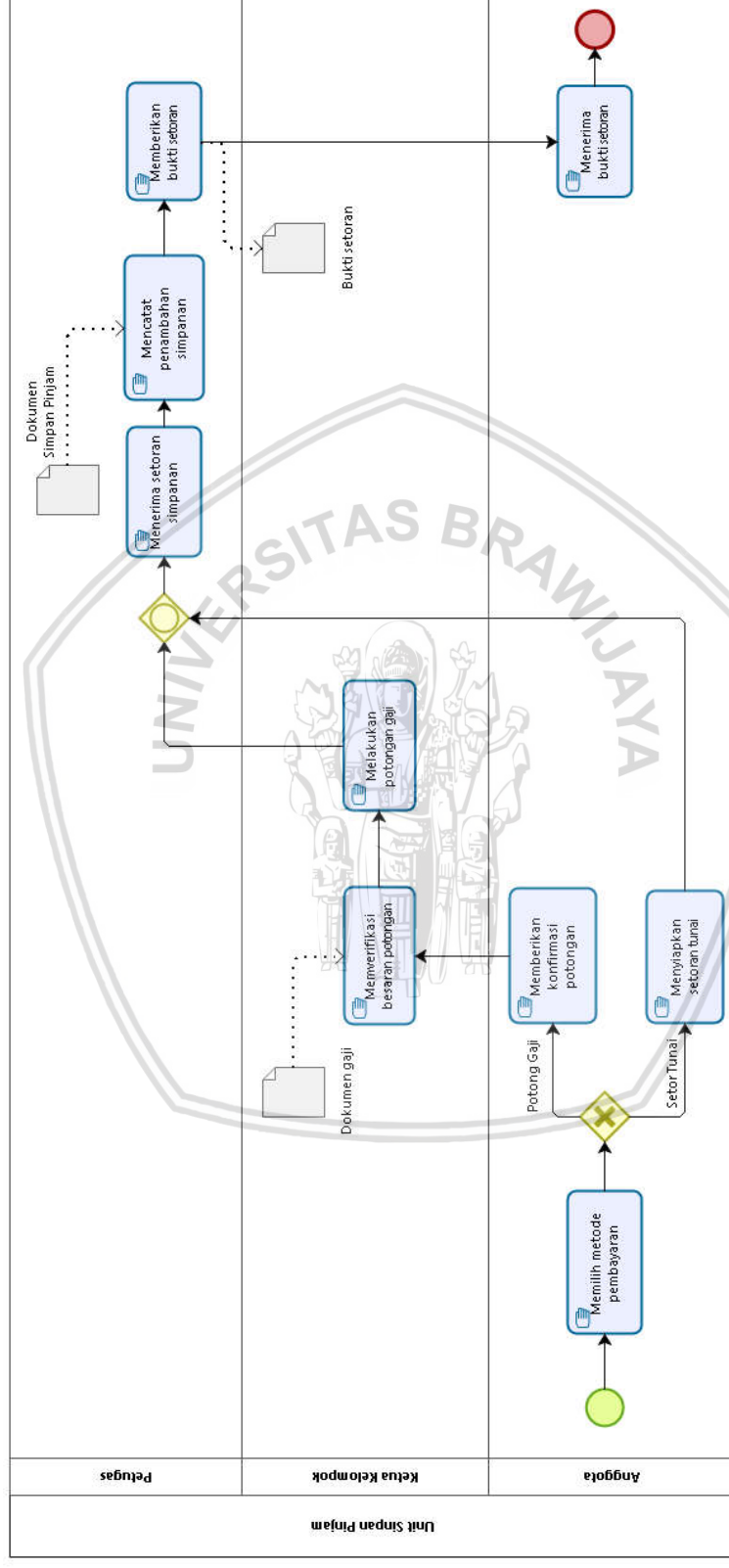
#### 4. Proses bisnis *As-Is* setoran

Pada gambar 4.4 menjelaskan proses menyetorkan pinjaman maupun simpanan yang dimulai dengan anggota memilih metode penyetoran yaitu potongan gaji atau setoran langsung. Ketika anggota memilih metode pemotongan gaji, maka anggota akan menginformasikan besaran potongan gaji yang akan di setorkan. Ketua kelompok akan melihat dokumen gaji sebelum melakukan pemotongan gaji. Setelah proses selesai maka proses setelahnya sama dengan metode setor tunai yaitu mendatangi kantor KPRI untuk menyetorkan uang kemudian menyimpan data ke dokumen simpan pinjam. Proses terakhir adalah memberikan bukti setoran





### Gambar 4.3 As-Is pengajuan pinjaman



Gambar 4.4 As-is setoran

#### 4.1.2 Analisis Permasalahan

Analisis permasalahan dilakukan untuk mengetahui pengaruh permasalahan yang ada dan cara menyelesaikan permasalahan tersebut. Analisis permasalahan berdasar pada hasil wawancara, observasi dan permodelan proses bisnis *as-is* pada Unit Simpan Pinjam KPRI “GURU” Kec. Sumbermanjing Wetan.

**Table 4.1 Analisis Masalah Pengajuan Pinjaman**

Masalah	<ol style="list-style-type: none"> <li>1. Anggota harus datang ke kantor hanya sekedar untuk mendapatkan informasi simpanan atau pinjaman yang dimiliki.</li> <li>2. Anggota harus datang ke kantor setidaknya 3 kali selama proses pengajuan pinjaman berlangsung.</li> <li>3. Petugas akan melakukan perhitungan berulang ketika memberikan informasi peminjaman.</li> <li>4. Ketika terdapat banyak permohonan pinjaman akan terdapat antrian.</li> <li>5. Ketua kelompok tidak dapat mengetahui hasil dari pengajuan.</li> </ol>
Mempengaruhi	Anggota, ketua kelompok dan petugas
Dampak	<ol style="list-style-type: none"> <li>1. Anggota akan mengeluarkan banyak waktu dan biaya dalam perjalanan.</li> <li>2. Anggota akan beralih kepada koperasi penyedia jasa pinjaman lain yang lebih mudah dijangkau.</li> <li>3. Petugas membutuhkan waktu yang lama untuk menjelaskan setiap perkiraan pinjaman kepada anggota.</li> <li>4. Ketua kelompok harus menunggu atau menanyakan hasil dari pengajuan.</li> </ol>
Solusi	<ol style="list-style-type: none"> <li>1. Membuat sebuah sistem yang dapat memberikan informasi perhitungan dari setiap estimasi pinjaman secara otomatis.</li> </ol>

	2. Membuat sebuah sistem informasi pengajuan pinjaman 3. Merubah alur pengajuan sehingga anggota hanya 1 kali datang ke kantor. 4. Merubah alur melengkapi persyaratan menjadi ketika pengajuan diterima
--	--

Table 4.2 menjelaskan tentang analisis permasalahan yang ada saat ini dan waktu yang diperlukan dalam melakukan proses tersebut. Permasalahan yang ada merupakan masalah yang ditemukan pada proses bisnis *as-is* dan waktu yang diperlukan merupakan estimasi berdasarkan observasi yang telah dilakukan.

**Table 4.2 Analisis permasalahan dan waktu yang dibutuhkan**

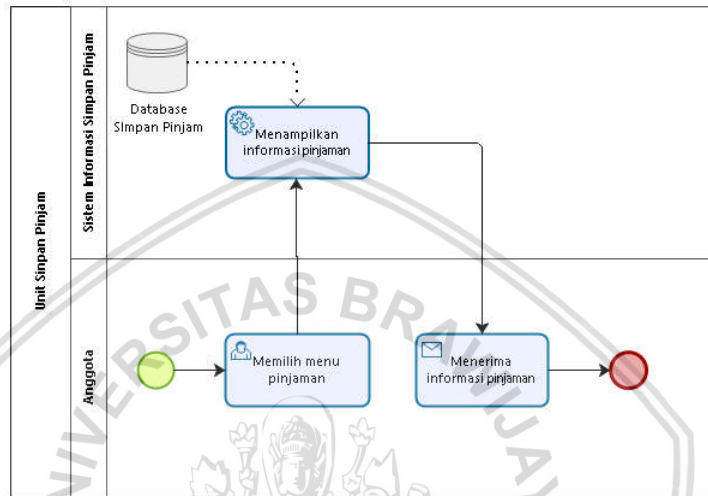
No	Identifikasi Masalah	Waktu yang diperlukan
1	Anggota harus datang ke kantor untuk mendapatkan informasi simpanan atau pinjaman yang dimiliki	45 menit
2	Anggota harus datang ke kantor koperasi untuk meminta form dan mendapatkan informasi estimasi kredit.	45 menit
3	Anggota menunggu hasil perhitungan estimasi	15 menit
4	Anggota harus kembali untuk memenuhi persyaratan.	45 menit
5	Setelah form permohonan lengkap anggota harus kembali mengembalikan kepada petugas unit simpan pinjam	45 menit
6	Setelah menunggu hasil permohonan yang dapat dilihat setiap tanggal 20, anggota harus datang ke kantor untuk melihat informasi tersebut.	45 menit

#### 4.1.3 Permodelan Proses Bisnis *To-be*

Permodelan proses bisnis *to-be* yang merupakan proses bisnis usulan sebagai solusi dari proses bisnis *as-is*. Usulan perbaikan proses bisnis berdasarkan pada analisis permasalahan yang ada pada proses bisnis yang ada saat ini. Validasi proses bisnis *to-be* dilakukan kepada pemangku kepentingan untuk memastikan proses bisnis tersebut nantinya dapat dijalankan. Berikut merupakan hasil permodelan pada fase inepsi atau fase pertama dalam *RUP*.

### 1. Proses bisnis *to-be* mendapatkan informasi simpanan

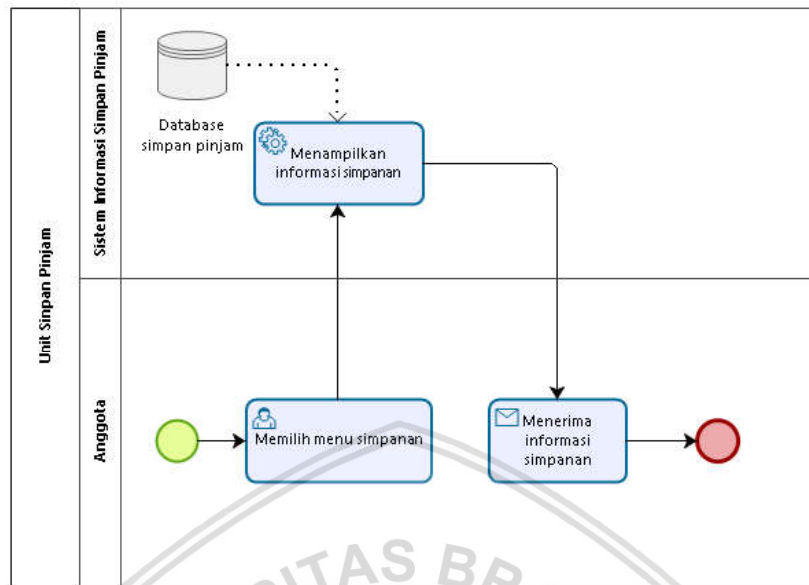
Gambar 4.5 merupakan proses bisnis *to-be* dari proses mendapatkan informasi simpanan. Perubahan dari proses bisnis *as-is* adalah terdapat pada jenis proses yang ada serta bagian yang berkaitan. Pada proses bisnis *as-is* anggota berinteraksi dengan petugas namun pada proses bisnis *to-be* anggota mendapatkan informasi dari sistem informasi.



Gambar 4.5 *To-Be* informasi pinjaman

### 2. Proses bisnis *to-be* mendapatkan informasi simpanan

Pada gambar 4.6 merupakan notasi BPMN dari proses bisnis *to-be* untuk mendapatkan informasi simpanan. Perubahan proses bisnis yang terjadi adalah perubahan jenis *task* dari *manual task* menjadi *user task* karena terdapat perubahan petugas yang digantikan dengan sistem. Perubahan dari petugas menjadi sistem juga merubah sumber data dokument menjadi database.

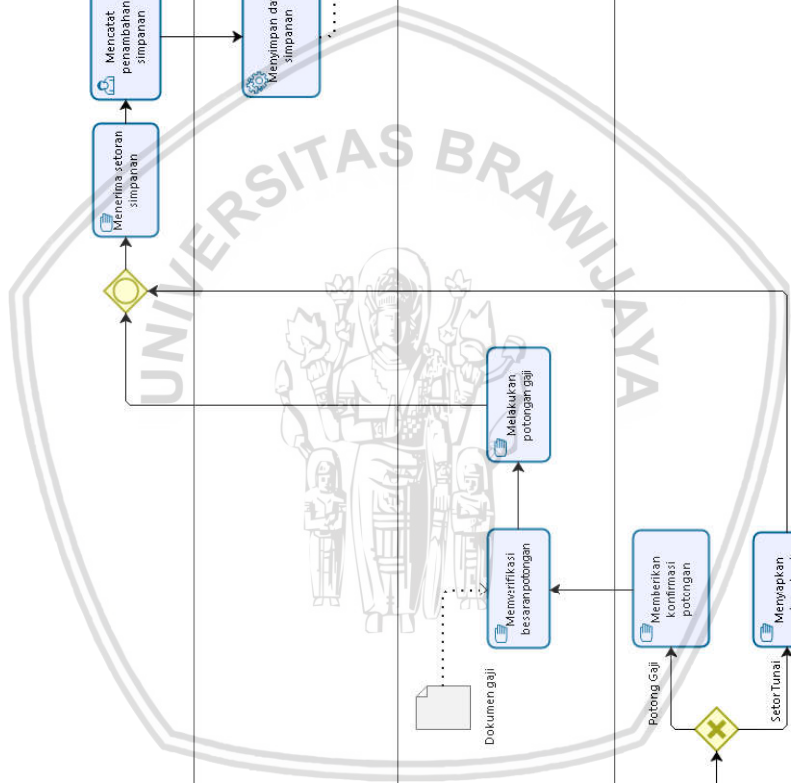


**Gambar 4.6 To-Be informasi simpanan**

### 3. Proses bisnis *to-be* setoran

Pada proses bisnis setoran terdapat perubahan pada proses pencatatan data simpan pinjam. Proses pencatatan perubahan data simpan pinjam pada proses bisnis *as-is* dimana data disimpan dalam bentuk dokumen dan dikejakan secara manual berubah menjadi *database* seperti yang terdapat pada gambar 4.7. Tidak terdapat perubahan pada proses memberikan bukti setoran dikarenakan ketika sistem yang baru diterapkan pemberian bukti setoran akan tetap dicetak sebagai bukti fisik setoran.



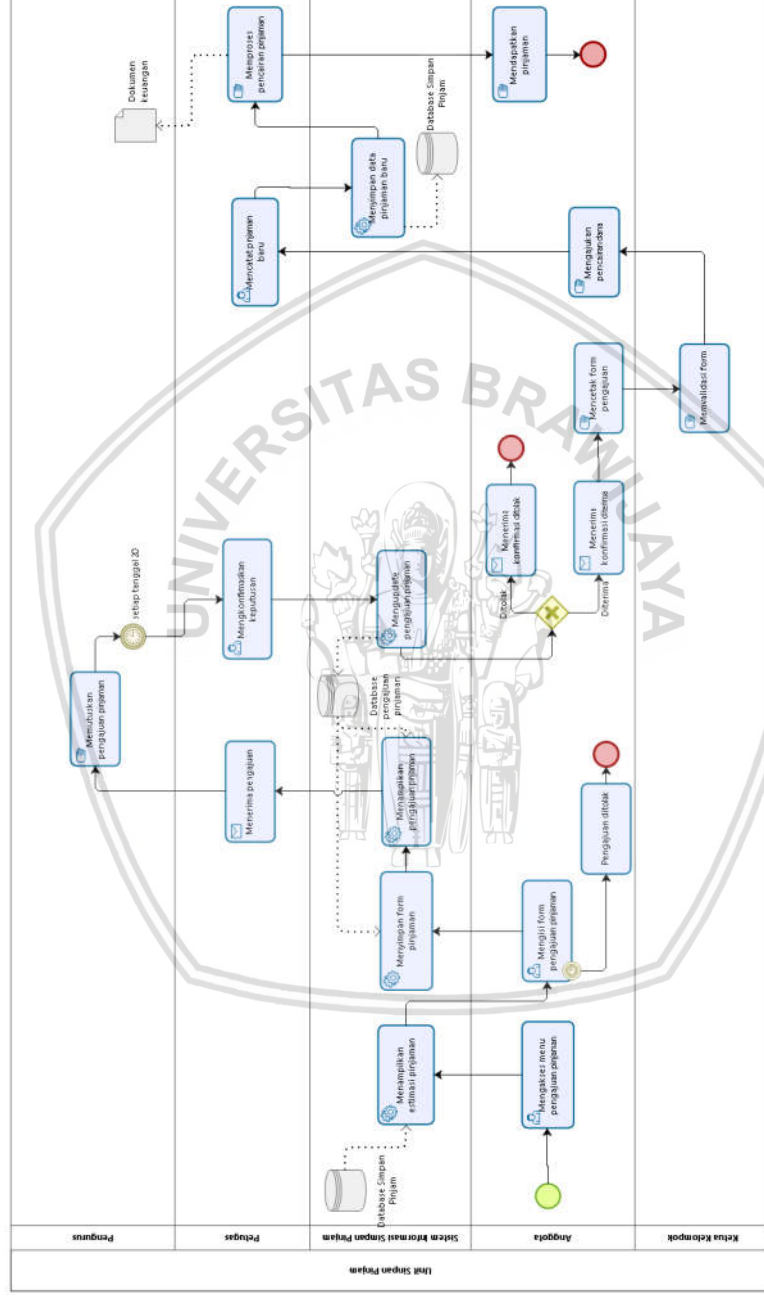


50

#### 4. Proses bisnis *to-be* pengajuan pinjaman

Pada proses bisnis pengajuan pinjaman terdapat perubahan signifikan dari proses bisnis yang saat ini ada. Seperti yang digambarkan pada Gambar 4.8, pada proses bisnis *to-be* terdapat sistem informasi sehingga beberapa *task* yang awalnya dilakukan oleh petugas kini dapat dilakukan otomatis oleh sistem. Selain itu terdapat perubahan urutan pada task validasi oleh ketua kelompok, perubahan ini bertujuan untuk menimalkan proses anggota datang ke kantor KPRI.





### Gambar 4.8 To-Be pengajuan pinjaman

#### 4.1.4 Analisa Perbaikan Proses bisnis

Analisa perbaikan proses bisnis untuk mengetahui proses bisnis *to-be* atau proses bisnis usulan memberikan solusi dan peningkatan dari proses bisnis *as-is*. Analisa menggunakan perbandingan waktu dari premasalahan yang ada pada proses bisnis *as-is* dengan peroses bisnis *to-be*. Waktu yang didapatkan berdasar pada perkiraan rata-rata estimasi waktu yang diperlukan untuk aktivitas – aktivitas tersebut. Berikut ini merupakan analisis perbaikan proses bisnis:

**Table 4.3 Perbandingan Proses Bisnis**

No	Proses bisnis <i>as-is</i>	Proses bisnis <i>to-be</i>	Waktu lama	Perkiraan waktu baru
1	Anggota harus datang ke kantor untuk mendapatkan informasi simpana atau pinjaman yang dimiliki	Anggota dapat melihat informasi melalui sistem informasi.	45 menit	5 menit
2	Anggota harus datang ke kantor koperasi untuk meminta form dan mendapatkan informasi estimasi kredit.	Anggota dapat melihat informasi dan mendapatkan form pengajuan melalui sistem informasi.	45 menit	10 menit
3	Petugas melakukan perhitungan estimasi kredit berulang - ulang	Anggota dapat mendapatkan informasi estimasi melalui sistem informasi yang ada.	15 menit	5 menit
4	Anggota harus kembali untuk memenuhi persyaratan	Proses ini hilang karena anggota tidak perlu datang ke kantor untuk mendapatkan form dan informasi kredit.	45 menit	0 menit
5	Setelah form permohonan lengkap anggota harus kembali mengembalikan kepada petugas unit simpan pinjam	Anggota tidak perlu datang untuk mengembalikan form secara langsung ke kantor karena digantikan pengisian	45 menit	15 menit

No	Proses bisnis <i>as-is</i>	Proses bisnis <i>to-be</i>	Waktu lama	Perkiraan waktu baru
		melaui sistem yang ada.		
6	Setelah mengganggu hasil permohonan yang dapat dilihat setiap tanggal 20, anggota harus datang ke kantor untuk melihat informasi tersebut.	Anggota tidak perlu datang ke kantor KPRI untuk mengetahui hasil keputusan karena hasil keputusan dapat dilihat pada sistem informasi.	45 menit	10 menit

## 4.2 Analisis Kebutuhan

### 4.2.1 Identifikasi Pemangku Kepentingan

Tipe pemangku kepentingan adalah kumpulan pemangku kepentingan yang memiliki hubungan dengan sistem yang dibangun. Setiap pemangku kepentingan dalam sistem informasi Unit Simpan Pinjam memiliki karakteristik yang berbeda-beda seperti berikut :

**Table 4.4 Tabel Kebutuhan Pemangku Kepentingan**

No	Tipe Pemangku Kepentingan	Deskripsi Pemangku Kepentingan	Pemangku Kepentingan
1	Pengguna ( <i>User</i> )	Seseorang yang berinteraksi langsung dengan sistem. Dideskripsikan sebagai aktor dalam pada <i>use case</i> .	Petugas, Anggota Unit Simpan Pinjam
2	Pihak Berwenang	Pihak yang memiliki wewenang terhadap sistem yang dikembangkan, merupakan pihak yang mengerti permasalahan dan solusi serta memiliki otoritas terhadap sistem secara keseluruhan.	Unit Simpan Pinjam
3	Pengembang	Pihak yang bertanggung jawab dalam	Peneliti

		pengembangan sistem yang akan dikembangkan	
--	--	--	--

#### 4.2.2 Kebutuhan Pemangku Kepentingan

Kebutuhan pemangku kepentingan diperoleh dari analisis proses bisnis, wawancara serta observasi dengan pihak yang berwenang pada Unit Simpan Pinjam. Urutan prioritas digunakan untuk mengetahui kebutuhan apa yang menjadi fokus utama dalam pengembangan sistem. Dari hasil analisis permasalahan yang telah dilakukan, dimana permasalahan terbesar berada pada pemangku kepentingan anggota maka urutan prioritas utama kebutuhan sistem berdasar pada jenis aplikasi *customer-touching applications*. Tabel 4.6 menjelaskan kebutuhan pemangku kepentingan sistem informasi Unit Simpan Pinjam.

**Table 4.5 Table Kebutuhan Pemangku Kepentingan**

Kebutuhan	Pemangku Kepentingan	Prioritas	Situasi Saat ini	Solusi yang di tawarkan
Kemudahan anggota untuk mendapatkan informasi simpan pinjam	Anggota	Mo	Anggota harus datang dan meminta informasi kepada petugas	Informasi dapat diakses oleh anggota dengan mudah melalui sistem.
Kemudahan untuk melakukan pengajuan pinjaman	Anggota	Mo	Pengajuan kredit saat ini mewajibkan anggota untuk datang ketika meminta form pengajuan dan informasi kredit	Pengisian Form pengajuan kredit dapat dilakukan lewat sistem yang akan dibangun.
Menyediakan fasilitas mengelola data simpan pinjam seluruh anggota	Petugas	S	Seluruh aktifitas dalam pengolahan data masih bersifat manual.	Petugas dapat mengelola data menggunakan sistem sehingga lebih mudah.
Menyediakan fasilitas untuk mengelola	Petugas	S	Petugas harus membuka beberapa dokumen	Petugas dapat memperoleh seluruh informasi



Kebutuhan	Pemangku Kepentingan	Prioritas	Situasi Saat ini	Solusi yang di tawarkan
pengajuan pinjaman			untuk mengetahui rekam jejak dari pemohon pinjaman	secara otomatis ketika terdapat permohonan pinjaman.

#### 4.2.3 Identifikasi Pengguna

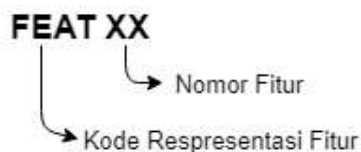
Perang pengguna merupakan peran yang dijalankan seseorang saat berinteraksi dengan sistem. Peran pengguna berfungsi untuk mengetahui seberapa jauh seorang pengguna dapat berinteraksi dengan sistem. Tabel merupakan peran pengguna di dalam sistem informasi Unit Simpan Pinjam

**Table 4.6 Tabel Peran Pengguna**

No	Peran Pengguna	Deskripsi
1	User	User merupakan pengguna sistem yang dapat melakukan <i>login</i> sesuai dengan hak akses
2	Petugas	Merupakan pengguna yang dapat mengelola data simpan pinjam dan pengajuan pinjaman.
3	Anggota	Merupakan pengguna yang dapat melihat informasi simpan pinjam dan mengajukan pinjaman pada sistem.

#### 4.2.4 Identifikasi Fitur

Fitur merupakan suatu kemampuan sistem yang dapat menyajikan nilai bagi pengguna. Fitur akan dinilai berdasarkan prioritas untuk mengetahui fitur mana yang menjadi fokus dalam pengembangan. Fitur sistem diambil berdasarkan kebutuhan pemangku kepentingan dan pengguna. Table merupakan fitur dari sistem informasi *Customer - Touching Applications Unit Simpan Pinjam* pada unit simpan pinjam KPRI.



Gambar 4.9 Kodefikasi Fitur

Table 4.7 Tabel Fitur Pengguna

Kode Fitur	Fitur	Deskripsi	Prioritas
FEAT 01	Autentifikasi	Sistem menyediakan fasilitas bagi user untuk mengakses sistem sesuai hak akses.	Mo
FEAT 02	Melihat informasi simpanan	Sistem menyediakan fasilitas untuk melihat informasi simpanan	Mo
FEAT 03	Melihat informasi pinjaman	Sistem menyediakan fasilitas untuk melihat informasi pinjaman	Mo
FEAT 04	Melakukan pengajuan pinjaman	Sistem menyediakan fasilitas untuk melakukan pengajuan pinjaman	Mo
FEAT 05	Melakukan konfirmasi pinjaman	Sistem menyediakan fasilitas untuk melakukan konfirmasi pengajuan pinjaman	S
FEAT 06	Mengelola Data simpan pinjam	Sistem menyediakan fasilitas untuk mengakomodasi perubahan data	S
FEAT 07	Mengelola data anggota	Sistem menyediakan fasilitas untuk mengelola data anggota	S
FEAT 08	<i>Compatibility</i>	Sistem dapat diakses dari berbagai browser	Mo

#### 4.2.5 Spesifikasi Kebutuhan

Spesifikasi kebutuhan dalam sistem *Customer - Touching Applications Unit Simpan Pinjam* berdasarkan pada analisa fitur sistem dibagi menjadi kebutuhan fungsional dan kebutuhan non fungsional. Termasuk dalam fase inepsi dalam

metode *RUP* sehingga masih dapat berubah ketika terjadi perubahan pada fase selanjutnya.



**Gambar 4.10 Kodefikasi Kebutuhan**

#### 4.2.5.1 Kebutuhan Fungsional

Kebutuhan fungsional merupakan suatu aksi yang dapat dilakukan oleh sistem. Kebutuhan fungsional sistem *Customer - Touching Applications Unit Simpan Pinjam* sendiri dikelompokkan berdasarkan kebutuhan tiap pengguna, sebagai berikut:

##### 1. User

**Table 4.8 Kebutuhan Fungsional User**

Kode Fitur	Kode Fungsi	Nama Fungsi	Deskripsi
FEAT 01	SIKPRI-KF-01	Log In	Sistem dapat mengotentifikasi pengguna berdasar pada data yang telah tersimpan.
	SIKPRI-KF-02	Log Out	Sistem dapat menghentikan <i>session</i> pengguna.

Tidak terdapat perubahan kebutuhan fungsional pada fase berikutnya yaitu fase elaborasi. Pemangku kepentingan telah setuju dengan perancangan yang telah dibuat. Sehingga table 4.8 digunakan sebagai acuan kebutuhan fungsional dari *user* pada pengembangan sistem informasi ini.

##### 2. Anggota

**Table 4.9 Kebutuhan Fungsional Anggota (Fase Insepsi)**

Kode Fitur	Kode Fungsi	Nama Fungsi	Deskripsi
FEAT 02	SIKPRI-KF-03	Melihat informasi simpanan.	Sistem dapat digunakan untuk dapat melihat informasi simpanan yang dimiliki.
FEAT 03	SIKPRI-KF-04	Melihat informasi pinjaman	Sistem dapat digunakan untuk melihat informasi

			detail simpanan yang dimiliki.
FEAT 04	SIKPRI-KF-05	Mengajukan pinjaman	Sistem dapat digunakan untuk melakukan pengajuan pinjaman

Table 4.9 merupakan daftar kebutuhan fungsional dari anggota. Terdapat perubahan kebutuhan fungsional anggota pada fase Elaborasi. Perubahan tersebut antara lain penambahan kebutuhan fungsional melihat detail simpanan, melihat detail pinjaman dan memisahkan perhitungan estimasi pinjaman dari fungsi mengajukan pinjaman. Hal ini dilakukan agar pengguna dapat melakukan perhitungan estimasi tanpa harus mengajukan pinjaman. Selain itu juga terdapat penambahan fungsi unduh form pengajuan yang termasuk kedalam fitur pengajuan pinjaman. Perubahan terjadi karena anggota membutuhkan kemudahan untuk mendapatkan file form pengajuan pinjaman untuk memenuhi persyaratan. Terdapat penambahan kebutuhan fungsional lainnya seperti mengubah *password* untuk memberikan privasi kepada pengguna. Pada fase konstruksi terdapat penambahan kebutuhan yaitu melihat detail pengajuan yang telah dilakukan. Sehingga kebutuhan fungsional anggota yang disepakati untuk diterapkan dalam sistem terdapat pada table 4.10.

**Table 4.10 Tambahan Kebutuhan Fungsional Anggota (Fase Konstruksi)**

Kode Fitur	Kode Fungsi	Nama Fungsi	Deskripsi
FEAT 02	SIKPRI-KF-20	Melihat detail simpanan	Sistem dapat digunakan untuk melihat informasi detail simpanan yang dimiliki.
FEAT 03	SIKPRI-KF-21	Melihat detail pinjaman	Sistem dapat digunakan untuk melihat informasi detail simpanan yang dimiliki.
FEAT 04	SIKPRI-KF-22	Menghitung estimasi	Sistem dapat digunakan untuk menghitung perkiraan potongan bulanan dan bunga dari pinjaman yang akan diajukan.
	SIKPRI-KF-23	Mengunduh form pengajuan pinjaman	Sistem dapat digunakan untuk mengunduh file form pengajuan
	SIKPRI-KF-25	Melihat detail pengajuan	Sistem dapat digunakan untuk mengetahui detail

			pengajuan pinjaman yang telah dilakukan
FEAT 07	SIKPRI-KF-24	Mengubah <i>password</i>	Sistem dapat digunakan untuk mengubah <i>password</i> pengguna.

### 3. Petugas

**Table 4.11 Kebutuhan Fungsional Petugas**

Kode Fitur	Kode Fungsi	Nama Fungsi	Deskripsi
FEAT 05	SIKPRI-KF-06	Konfirmasi pengajuan	Sistem dapat digunakan untuk mengkonfirmasi permohonan pinjaman
	SIKPRI-KF-07	Melihat data pengajuan	Sistem dapat digunakan untuk melihat data dari pengajuan yang ada
FEAT 06	SIKPRI-KF-08	Melihat data simpanan	Sistem dapat digunakan untuk menampilkan data simpanan bagi petugas
	SIKPRI-KF-09	Menambah data simpanan	Sistem dapat digunakan untuk menambah data simpanan
	SIKPRI-KF-10	Mengubah data simpanan	Sistem dapat digunakan untuk melakukan perubahan data simpanan
	SIKPRI-KF-11	Menghapus data simpanan	Sistem dapat digunakan untuk menghapus data simpanan
	SIKPRI-KF-12	Melihat data pinjaman	Sistem dapat digunakan untuk menampilkan data pinjaman
	SIKPRI-KF-13	Menambah data pinjaman	Sistem dapat digunakan untuk menambah data pinjaman
	SIKPRI-KF-14	Mengubah data pinjaman	Sistem dapat digunakan untuk melakukan perubahan data pinjaman

Kode Fitur	Kode Fungsi	Nama Fungsi	Deskripsi
	SIKPRI-KF-15	Menghapus data pinjaman	Sistem dapat digunakan untuk menghapus data pinjaman
FEAT 07	SIKPRI-KF-16	Melihat data anggota	Sistem dapat digunakan untuk menampilkan data anggota
	SIKPRI-KF-17	Menambah data anggota	Sistem sistem dapat digunakan untuk menambah data anggota
	SIKPRI-KF-18	Mengubah data anggota	Sistem dapat digunakan untuk melakukan perubahan data anggota
	SIKPRI-KF-19	Menghapus data anggota	Sistem dapat digunakan untuk menghapus data anggota

Table 4.11 merupakan kebutuhan fungsional dari aktor petugas. Terdapat perubahan pada kebutuhan fungsional petugas pada fase elaborasi. Perubahan tersebut berupa pengurangan kebutuhan yaitu menambah data simpanan dan menghapus data simpanan. Penghapusan kebutuhan tersebut dikarenakan pada unit simpan pinjam penambahan atau penghapusan data simpanan bergabung dengan penambahan atau penghapusan data anggota. Sehingga untuk kebutuhan fungsional petugas menambah dan menghapus data simpanan dihapus dari kebutuhan fungsional sistem yang dikembangkan.

#### 4.2.5.2 Kebutuhan Non Fungsional

Kebutuhan non fungsional merupakan kebutuhan yang menjelaskan kondisi sistem atau bagaimana sistem akan berjalan. Analisa kebutuhan non fungsional yang merupakan bagian dari fase inisiasi pada perancangan sistem informasi simpan pinjam yang akan dikembangkan tidak terdapat perubahan pada selanjutnya atau fase elaborasi.

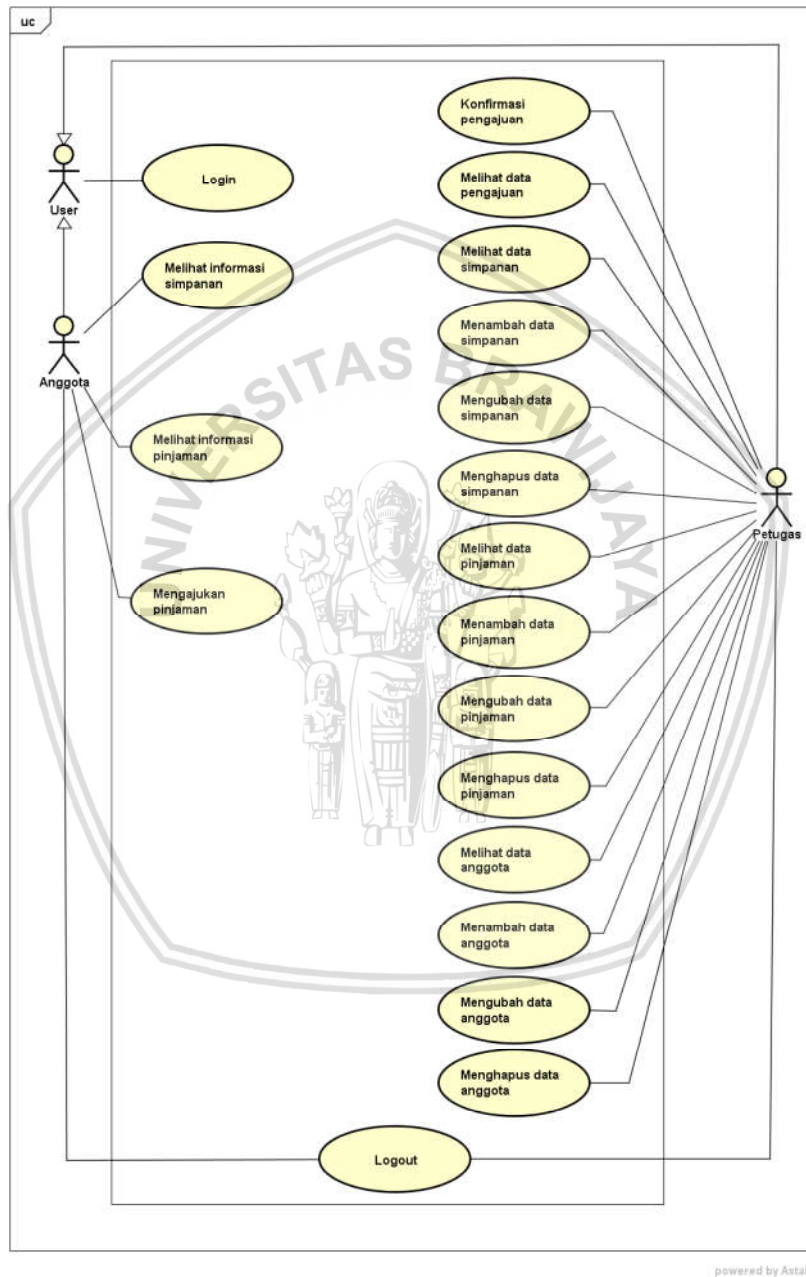
**Table 4.12 Kebutuhan Non Fungsional**

Kode Fitur	Kode Fungsi	Nama Fungsi	Deskripsi
FEAT 08	SIKPRI-KN-01	<i>Compatibility</i>	Sistem dapat berjalan diberbagai <i>browser</i> yang berbeda.



### 4.3 Permodelan Use Case Diagram

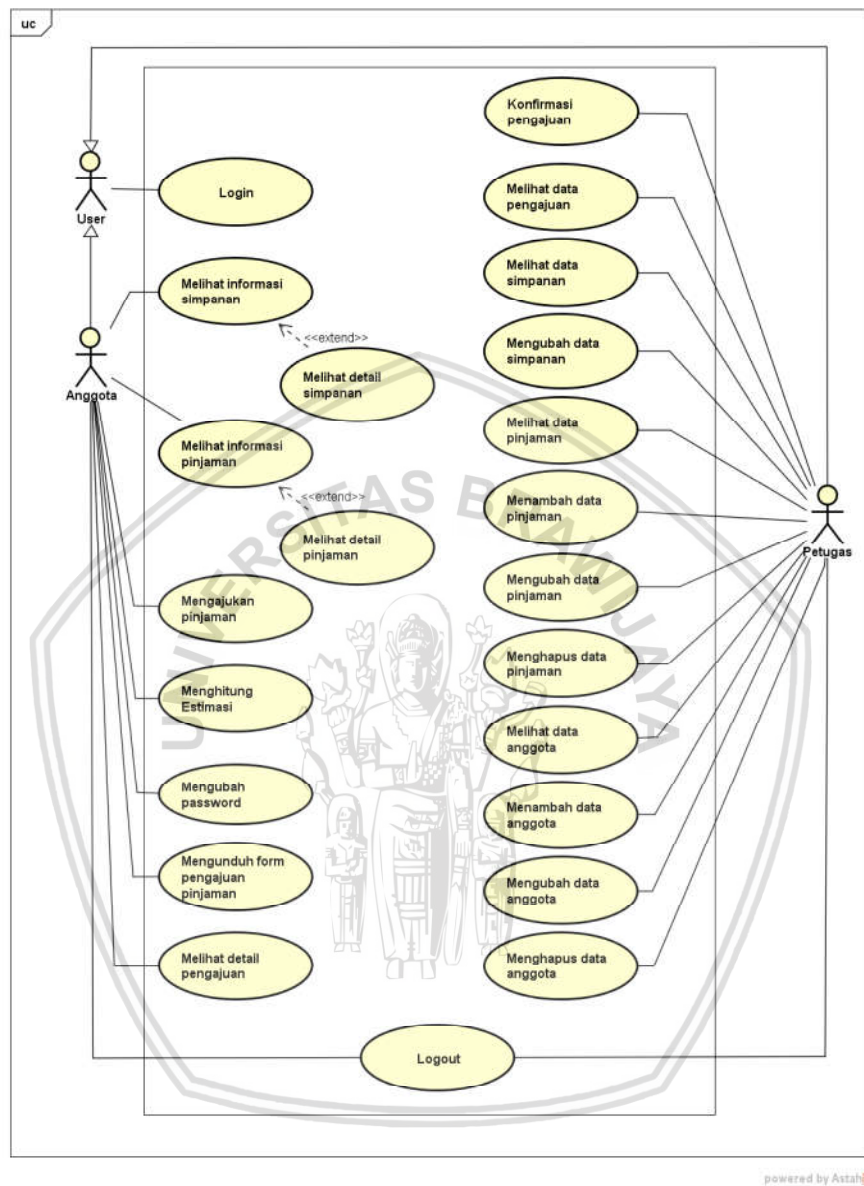
*Use case diagram* merupakan diagram *Unified Modelling Language (UML)* yang merepresentasikan aktor-aktor yang terlibat dalam sistem *Customer - Touching Applications Unit Simpan Pinjam* beserta peran (berupa *use case*) yang mengacu pada kebutuhan fungsionalitas sistem.



**Gambar 4.11 Use Case Diagram Fase Insepsi**

Gambar 4.11 merupakan *use case diagram* pada awal fase insepsi. Dari *use case diagram* awal tersebut terdapat perubahan berupa penambahan dan pengurangan *use case*. Perubahan *use case diagram* berdasar pada perubahan

kebutuhan fungsional yang ada. Sehingga pada *use case diagram* yang diterapkan seperti pada gambar 4.12.



Gambar 4.12 *Use Case* Fase Konstruksi

#### 4.3.1 Skenario *Use Case*

Skenario berikut merupakan skenario yang menjabarkan masing –masing *use case* pada gambar 4.10 mengenai *use case diagram* dalam sistem informasi *customer touching applications* unit simpan pinjam KPRI :

**Table 4.13 Skenario Use Case Login**

<i>Flow of Events</i> untuk <i>use case</i> melakukan login	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor masuk kedalam sistem
<i>Actor</i>	Pengguna
<i>Pre-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor telah terdaftar dan memiliki akun dalam sistem</li> <li>2. Aktor berada pada halaman <i>login</i></li> </ol>
<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor menginputkan <i>username</i> dan <i>password</i></li> <li>2. Aktor memilih login</li> <li>3. Sistem akan melakukan autentifikasi</li> <li>4. Sistem menampilkan halaman awal</li> </ol>
<i>Alternative Flows</i>	<p>A1. <i>Username</i> dan <i>password</i> salah</p> <p>Pada tahap sistem menampilkan halaman awal, jika <i>username</i> dan <i>password</i> tidak sesuai maka sistem akan menampilkan halaman login kembali.</p>
<i>Post-conditions</i>	Aktor masuk kedalam sistem dan di- <i>redirect</i> menuju halaman awal.

Table 4.13 menjelaskan skenario bagaimana aktor dapat masuk kedalam sistem. Aktor yang berperan dalam *use case login* adalah pengguna. *Use case* ini dapat dilakukan ketika aktor telah terdaftar dan berada pada halaman login masing – masing. Langkah – langkah dalam *use case* ini adalah aktor meninputkan *username* dan *password* dan kemudian menekan tombol *login* dan sistem akan melakukan otentifikasi. Akhir dari *use case* ini adalah aktor dapat masuk kedalam sistem dengan ditandai berada pada halaman awal masing – masing.

**Table 4.14 Skenario Use Case melihat informasi simpanan**

<i>Flow of Events</i> untuk <i>use case</i> melakukan melihat informasi simpanan	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat mendapatkan informasi simpanan yang dimiliki
<i>Actor</i>	Anggota
<i>Pre-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor telah masuk kedalam sistem</li> </ol>
<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih fungsi lihat simpanan</li> <li>2. Sistem menampilkan informasi simpanan yang dimiliki aktor</li> </ol>
<i>Alternative Flows</i>	-

<i>Post-conditions</i>	Aktor mendapat informasi simpanan yang dimiliki
------------------------	---

Table 4.14 menjelaskan skenario bagaimana aktor dapat mendapatkan informasi simpan yang dimiliki. Aktor yang berperan dalam *use case* ini adalah anggota. *Use case* ini dapat dilakukan ketika aktor telah masuk kedalam sistem. Akhir dari *use case* ini adalah aktor berada pada halaman simpanan dan mendapatkan informasi simpanan yang dimiliki.

**Table 4.15 Skenario *Use Case* melihat pinjaman**

Flow of Events untuk <i>use case</i> melakukan melihat informasi pinjaman	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat mendapatkan informasi pinjaman yang dimiliki
<i>Actor</i>	Anggota
<i>Pre-conditions</i>	1. Aktor telah masuk kedalam sistem
<i>Basic flow of Events</i>	1. Aktor memilih fungsi lihat informasi pinjaman 2. Sistem menampilkan informasi pinjaman yang dimiliki aktor
<i>Alternative Flows</i>	-
<i>Post-conditions</i>	1. Aktor berada pada halaman simpanan 2. Aktor mendapat informasi simpanan yang dimiliki

Table 4.15 menjelaskan skenario bagaimana aktor dapat mendapatkan informasi pinjaman yang dimiliki. Aktor yang berperan dalam *use case* ini adalah anggota. *Use case* ini dapat dilakukan ketika aktor telah masuk kedalam sistem. Akhir dari *use case* ini adalah aktor berada pada halaman pinjaman dan mendapatkan informasi pinjaman yang dimiliki

**Table 4.16 Skenario *Use Case* mengajukan pinjaman (Fase Insepsi)**

Flow of Events untuk <i>use case</i> melakukan mengajukan pinjaman	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat melakukan pengajuan pinjaman
<i>Actor</i>	Anggota
<i>Pre-conditions</i>	1. Aktor telah masuk kedalam sistem 2. Aktor berada dalam halaman pengajuan pinjaman
<i>Basic flow of Events</i>	1. Aktor mengisi form perhitungan estimasi 2. Sistem menampilkan hasil estimasi pinjaman 3. Aktor memilih fungsi isi form pengajuan pinjaman

	<ol style="list-style-type: none"> <li>4. Sistem menampilkan form pengajuan pinjaman dan data aktor</li> <li>5. Aktor mengisi formulir pengajuan</li> <li>6. Sistem menyimpan data</li> </ol>
<i>Alternative Flows</i>	<p>A1 Input tidak sesuai format</p> <p>Pada tahapan sistem menampilkan hasil perhitungan, jika nilai pinjaman bernilai minus atau kosong maka sistem akan menampilkan notifikasi data yang dimasukkan tidak valid.</p> <p>A2 Form pengajuan tidak lengkap</p> <p>Pada tahapan sistem menampilkan halaman pengajuan, jika dalam pengisian form terdapat data yang tidak lengkap maka sistem akan menampilkan halaman form kembali.</p>
<i>Post-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor berhasil mengajukan pinjaman</li> <li>2. Data pengajuan pinjaman bertambah</li> </ol>

Table 4.16 merupakan skenario use case mengajukan pinjaman yang dibuat pada fase inepsi. Pada literasi kedua fase elaborasi terdapat perubahan pada kebutuhan sistem, yaitu dipisahkannya aktivitas menghitung estimasi pinjaman dari *use case* mengajukan pinjaman. Sehingga terdapat perubahan pada skenario *use case* mengajukan pinjaman yang dibuat pada fase inepsi.

**Table 4.17 Skenario *Use Case* mengajukan pinjaman (Fase Elaborasi)**

<i>Flow of Events</i> untuk <i>use case</i> melakukan mengajukan pinjaman	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat melakukan pengajuan pinjaman
<i>Actor</i>	Anggota
<i>Pre-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor telah masuk kedalam sistem</li> <li>2. Aktor berada dalam halaman pengajuan pinjaman</li> </ol>
<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih fungsi isi form pengajuan pinjaman</li> <li>2. Sistem menampilkan form pengajuan pinjaman dan data aktor</li> <li>3. Aktor mengisi formulir pengajuan</li> <li>4. Sistem menyimpan data</li> </ol>
<i>Alternative Flows</i>	A1 Form pengajuan tidak lengkap



	Pada tahapan sistem menampilkan halaman pengajuan, jika dalam pengisian form terdapat data yang tidak lengkap maka sistem akan menampilkan halaman form kembali.
<i>Post-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor berhasil mengajukan pinjaman</li> <li>2. Data pengajuan pinjaman bertambah</li> </ol>

Table 4.17 menjelaskan skenario bagaimana aktor dapat mengajukan pinjaman hal dari perubahan pada literasi ke 2 fase elaborasi. Aktor yang berperan dalam *use case* ini adalah anggota. *Use case* ini dapat dilakukan ketika aktor telah masuk kedalam sistem. Dalam *use case* ini terdapat *alternative flows* dimana jika pada form pengajuan tidak lengkap. Akhir dari *use case* ini adalah aktor berhasil mengajukan pengajuan pinjaman.

**Table 4.18 Skenario *Use Case* konfirmasi pinjaman**

<i>Flow of Events</i> untuk <i>use case</i> melakukan konfirmasi pinjaman	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat melakukan konfirmasi pengajuan pinjaman
<i>Actor</i>	Petugas
<i>Pre-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor telah masuk kedalam sistem</li> <li>2. Terdapat pengajuan pinjaman</li> <li>3. Aktor berada halaman pengajuan pinjaman</li> </ol>
<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih konfirmasi pengajuan pinjaman</li> <li>2. Sistem menyimpan hasil konfirmasi pinjaman</li> </ol>
<i>Alternative Flows</i>	-
<i>Post-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor berhasil melakukan konfirmasi</li> <li>2. Hasil pinjaman diterima atau ditolak</li> </ol>

Table 4.18 menjelaskan skenario bagaimana aktor dapat melakukan konfirmasi terhadap pengajuan pinjaman. Aktor yang berperan dalam *use case* ini adalah petugas. *Use case* ini dapat dilakukan ketika aktor telah masuk kedalam sistem dan terdapat pengajuan pinjaman. Akhir dari *use case* ini adalah aktor dapat melakukan konfirmasi pengajuan pinjaman.

**Table 4.19 Skenario *Use Case* melihat data simpanan**

<i>Flow of Events</i> untuk <i>use case</i> melihat data simpanan	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat melihat data simpanan
<i>Actor</i>	Petugas
<i>Pre-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor telah masuk kedalam sistem</li> </ol>



<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu simpanan</li> <li>2. Sistem menampilkan daftar simpanan</li> </ol>
<i>Alternative Flows</i>	-
<i>Post-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor berhasil melihat data simpanan</li> </ol>

Table 4.19 menjelaskan skenario bagaimana aktor dapat melihat data – data simpanan dari seluruh anggota unit simpan pinjam. Aktor yang berperan dalam *use case* ini adalah petugas. *Use case* ini dapat berjalan ketika aktor telah masuk kedalam sistem. Akhir yang diharapkan pada *use case* ini adalah aktor mendapatkan informasi simpanan setiap anggota.

**Table 4.20 Skenario *Use Case* Merubah Data Simpan**

<i>Flow of Events</i> untuk <i>use case</i> melakukan perubahan data simpan	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat melakukan perubahan data simpan
<i>Actor</i>	Petugas
<i>Pre-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor telah masuk kedalam sistem</li> <li>2. Terdapat data yang harus diubah</li> <li>3. Aktor berada pada halaman simpanan</li> </ol>
<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih fungsi <i>update</i></li> <li>2. Sistem menampilkan halaman perubahan data</li> <li>3. Aktor memilih jenis perubahan.</li> <li>4. Sistem menampilkan data berdasarkan jenis</li> <li>5. Aktor memilih simpan</li> <li>6. Sistem menyimpan dan menampilkan notifikasi status perubahan</li> </ol>
<i>Alternative Flows</i>	<p>A1 Jenis perubahan penarikan simpanan</p> <p>Pada tahap memilih jenis perubahan, jika aktor memilih penarikan maka aktor akan memasukkan besar penarikan.</p> <p>A2 Jenis perubahan simpanan manual</p> <p>Pada tahapan memilih jenis perubahan, jika aktor memilih perubahan berupa penambahan simpanan yang bersifat manual maka sistem akan menampilkan form besar perubahan.</p>
<i>Post-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor berhasil <i>update</i> data simpan</li> <li>2. Data simpan berubah</li> </ol>

Table 4.20 menjelaskan skenario bagaimana aktor dapat merubah data simpanan. Aktor yang berperan dalam *use case* ini adalah petugas. *Use case* ini dapat dilakukan ketika aktor telah masuk kedalam sistem. Terdapat 2 *alternative flows* yang dijalankan ketika jenis perubahan data berupa penarikan atau penambahan yang bersifat manual. Akhir dari *use case* ini adalah aktor dapat merubah data simpanan yang telah ada

**Table 4.21 Skenario Use Case Melihat Data Pinjaman**

<i>Flow of Events</i> untuk <i>use case</i> melihat data pinjaman	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat melihat data pinjaman
<i>Actor</i>	Petugas
<i>Pre-conditions</i>	1. Aktor telah masuk kedalam sistem
<i>Basic flow of Events</i>	1. Aktor memilih menu pinjaman 2. Sistem menampilkan daftar pinjaman 3. Aktor mendapat informasi pinjaman
<i>Alternative Flows</i>	
<i>Post-conditions</i>	1. Aktor berhasil melihat data pinjaman

Table 4.21 menjelaskan skenario bagaimana aktor dapat melihat data – data pinjaman dari seluruh anggota unit simpan pinjam. Aktor yang berperan dalam *use case* ini adalah petugas. *Use case* ini dapat berjalan ketika aktor telah masuk kedalam sistem. Akhir yang diharapkan pada *use case* ini adalah aktor mendapatkan informasi pinjaman setiap anggota

**Table 4.22 Skenario Use Case Menambah Pinjaman**

<i>Flow of Events</i> untuk <i>use case</i> melakukan penambahan data pinjam	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat melakukan penambahan data pinjam
<i>Actor</i>	Petugas
<i>Pre-conditions</i>	1. Aktor telah masuk kedalam sistem 2. Aktor berada pada halaman pinjaman
<i>Basic flow of Events</i>	1. Aktor memilih fungsi tambah pinjaman 2. Sistem menampilkan form data baru 3. Aktor mengisi form 4. Sistem menyimpan dan menampilkan notifikasi status penambahan
<i>Alternative Flows</i>	A1 Data yang diinputkan tidak lengkap  Pada tahapan sistem menyimpan dan menampilkan notifikasi, jika data yang diinputkan tidak lengkap

	maka sistem akan kembali menampilkan halaman form data pinjaman baru.
<i>Post-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor berhasil menambahkan data pinjam</li> <li>2. Data pinjam bertambah</li> </ol>

Table 4.22 menjelaskan skenario bagaimana aktor dapat melakukan penambahan data pinjaman. Aktor yang berperan dalam *use case* ini adalah petugas. *Use case* ini dapat dilakukan ketika aktor berada pada halaman pinjaman. Terdapat *alternative flows* yaitu ketika data yang form pinjaman baru tidak lengkap sehingga sistem akan menampilkan form tersebut kembali. Akhir dari *use case* ini adalah aktor dapat menambah data pinjaman baru.

**Table 4.23 Skenario Use Case Mengubah Data Pinjaman**

<i>Flow of Events</i> untuk <i>use case</i> melakukan perubahan data pinjaman	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat melakukan perubahan data pinjaman
<i>Actor</i>	Petugas
<i>Pre-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor telah masuk kedalam sistem</li> <li>2. Terdapat data yang harus diubah</li> <li>3. Aktor pada halaman pinjaman</li> </ol>
<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih fungsi <i>update</i> pada daftar pinjaman</li> <li>2. Sistem menampilkan form perubahan data</li> <li>3. Aktor memilih jenis setoran</li> <li>4. Sistem menyimpan dan menampilkan notifikasi status perubahan</li> </ol>
<i>Alternative Flows</i>	<p>A1 Aktor memilih setoran manual</p> <p>Pada tahap aktor memilih jenis setoran, jika aktor memilih jenis setoran manual maka sistem akan menampilkan form besar setoran yang dapat diisi oleh aktor.</p>
<i>Post-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor berhasil <i>update</i> data pinjaman</li> <li>2. Data pinjaman berubah</li> </ol>

Table 4.23 menjelaskan skenario bagaimana aktor dapat dapat merubah data pinjaman. Aktor yang berperan dalam *use case* ini adalah petugas. *Use ase* ini dapat dilakukan ketika aktor berada pada halaman pinjaman. Terdapat *alternative flows* ketika aktor memilih jenis setoran manual sehingga dapat diisi sesuai nominal yang dibayarkan. Akhir dari *use case* ini adalah aktor dapat merubah data pinjaman dan data pinjaman berubah.

**Table 4.24 Skenario Use Case Menghapus Data Pinjaman**

<i>Flow of Events</i> untuk <i>use case</i> melakukan menghapus data pinjaman	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat menghapus data pinjaman
<i>Actor</i>	Petugas

<i>Pre-conditions</i>	1. Aktor telah masuk kedalam sistem 2. Terdapat data yang harus dihapus
<i>Basic flow of Events</i>	1. Aktor memilih fungsi hapus 2. Sistem menampilkan konfirmasi penghapusan 3. Aktor memilih hapus 4. Sistem menghapus data
<i>Alternative Flows</i>	A1. Aktor batal hapus data  Pada tahap aktor memilih konfirmasi data, jika aktor memilih batal pada konfirmasi sistem kembali menampilkan daftar pinjaman.
<i>Post-conditions</i>	1. Aktor berhasil menghapus data 2. Data pinjam berkurang

Table 4.24 menjelaskan skenario bagaimana aktor dapat menghapus data pinjaman. Aktor yang berperan dalam *use case* ini adalah petugas. *Use case* ini dapat dilakukan ketika aktor berada pada halaman pinjaman. Akhir dari *use case* ini adalah aktor menghapus data pinjaman dan data pinjaman berkurang.

**Table 4.25 Skenario Use Case Melihat Data Anggota**

<i>Flow of Events</i> untuk <i>use case</i> melihat data anggota	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat melihat data anggota
<i>Actor</i>	Petugas
<i>Pre-conditions</i>	1. Aktor telah masuk kedalam sistem
<i>Basic flow of Events</i>	1. Aktor memilih menu anggota 2. Sistem menampilkan daftar anggota
<i>Alternative Flows</i>	-
<i>Post-conditions</i>	1. Aktor berhasil melihat data anggota

Table 4.25 menjelaskan skenario bagaimana aktor dapat melihat data anggota unit simpan pinjam KPRI. Aktor dalam *use case* ini adalah petugas dan syarat awal dari *use case* ini adalah petugas harus sudah masuk kedalam sistem. Akhir yang diharapkan dari *use case* ini adalah aktor dapat mendapatkan informasi data anggota.

**Table 4.26 Skenario Use Case Penambahan Data Anggota**

<i>Flow of Events</i> untuk <i>use case</i> melakukan penambahan data anggota	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat melakukan penambahan data anggota
<i>Actor</i>	Petugas
<i>Pre-conditions</i>	1. Aktor telah masuk kedalam sistem

	2. Aktor berada pada halaman anggota
<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih fungsi tambah anggota</li> <li>2. Sistem menampilkan form data baru</li> <li>3. Aktor mengisi form</li> <li>4. Sistem menyimpan penambahan data</li> </ol>
<i>Alternative Flows</i>	<p>A1 Data yang dimasukkan tidak lengkap</p> <p>Pada tahapan sistem menyimpan data baru, jika data yang dimasukkan tidak lengkap maka sistem akan menampilkan form tambah anggota lagi.</p> <p>A2 Data yang dimasukkan sama dengan data anggota non aktif</p> <p>Pada tahapan sistem menyimpan data baru, jika data yang dimasukkan sama dengan data anggota non aktif maka sistem akan menampilkan notifikasi.</p>
<i>Post-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor berhasil menambahkan data anggota</li> <li>2. Data bertambah bertambah</li> </ol>

Table 4.26 menjelaskan skenario bagaimana aktor dapat menambahkan data anggota pada *database* sistem. Aktor yang berperan dalam *use case* penambahan ini adalah petugas. Dalam *use case* ini terdapat 2 *alternative flows* yaitu ketika form tidak lengkap maka sistem akan menampilkan peringatan dan ketika data yang diinputkan sama dengan data anggota non aktif sistem akan menampilkan peringatan. Akhir dari *use case* ini adalah aktor dapat menambahkan data anggota baru.

**Table 4.27 Skenario Use Case Merubah Data Anggota**

<i>Flow of Events</i> untuk <i>use case</i> merubah data anggota	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat melakukan perubahan data anggota
<i>Actor</i>	Petugas
<i>Pre-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor telah masuk kedalam sistem</li> <li>2. Terdapat data yang harus diubah</li> <li>3. Aktor berada pada halaman anggota</li> </ol>
<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih fungsi <i>update</i></li> <li>2. Sistem menampilkan form perubahan data</li> <li>3. Aktor mengisi form perubahan data</li> <li>4. Sistem menyimpan dan menampilkan notifikasi status perubahan</li> </ol>
<i>Alternative Flows</i>	<p>A1 Data yang dimasukkan tidak lengkap</p> <p>Pada tahapan sistem menyimpan data perubahan, jika data yang dimasukkan tidak lengkap maka sistem akan menampilkan form perubahan anggota kembali.</p>

<i>Post-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor berhasil <i>update</i> data anggota</li> <li>2. Data anggota berubah</li> </ol>
------------------------	---

Table 4.27 menjelaskan skenario bagaimana aktor dapat melakukan perubahan data anggota. Aktor yang berperan dalam *use case* ini adalah petugas. Syarat awal dari *use case* ini adalah aktor telah masuk kedalam sistem. *Alternative flows* pada *use case* ini terdapat ketika form perubahan tidak lengkap maka sistem akan menampilkan peringatan dan form perubahan data kembali.

**Table 4.28 Skenario Use Case Menghapus Data Anggota**

<i>Flow of Events</i> untuk <i>use case</i> melakukan menghapus data anggota	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat menghapus data anggota
<i>Actor</i>	Petugas
<i>Pre-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor telah masuk kedalam sistem</li> <li>2. Terdapat data yang harus dihapus</li> </ol>
<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih data anggota yang dihapus</li> <li>2. Sistem merubah status anggota menjadi non aktif dan menghapus simpanan anggota</li> </ol>
<i>Alternative Flows</i>	-
<i>Post-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor berhasil menghapus data</li> <li>2. Data anggota aktif berkurang</li> </ol>

Table 4.28 menjelaskan skenario bagaimana aktor dapat menghapus data anggota. Aktor yang berperan dalam *use case* ini adalah petugas dan dengan syarat awal petugas telah masuk kedalam sistem. Akhir dari *use case* ini adalah aktor dapat menghapus data dari anggota aktif.

**Table 4.29 Skenario Use Case Logout**

<i>Flow of Events</i> untuk <i>use case</i> logout	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat keluar dari sistem
<i>Actor</i>	Petugas dan Anggota
<i>Pre-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor telah masuk kedalam sistem</li> </ol>
<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih fungsi logout</li> <li>2. Sistem menampilkan halaman <i>login</i></li> </ol>
<i>Alternative Flows</i>	-
<i>Key Scenarios</i>	Aktor berhasil keluar dari sistem
<i>Post-conditions</i>	Aktor keluar dari sistem

Table 4.29 menjelaskan skenario bagaimana aktor dapat keluar dari sistem. Aktor yang berperan dalam *use case* ini adalah petugas dan anggota. *Use case* ini



dapat dilakukan ketika aktor telah masuk kedalam sistem. Akhir dari *use case* ini adalah aktor keluar dari sistem.

**Table 4.30 Skenario Use Case Melihat Detail Simpanan (Tambahan dari Fase Elaborasi)**

<i>Flow of Events</i> untuk <i>use case</i> melakukan melihat detail simpanan	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat mendapatkan detail dari simpanan yang dimiliki
<i>Actor</i>	Anggota
<i>Pre-conditions</i>	1. Aktor telah masuk kedalam sistem 2. Aktor berada pada halaman simpanan
<i>Basic flow of Events</i>	1. Aktor memilih fungsi lihat detail simpanan 2. Sistem menampilkan data detail simpanan yang dimiliki aktor
<i>Alternative Flows</i>	-
<i>Post-conditions</i>	Aktor mendapat informasi detail simpanan yang dimiliki

Table 4.30 menjelaskan skenario bagaimana aktor dapat mendapatkan detail dari simpanan yang dimiliki. Aktor yang berperan dalam *use case* ini adalah anggota. *Use case* ini dapat dilakukan ketika aktor telah masuk kedalam sistem dan berada pada halaman simpanan. Akhir dari *use case* ini adalah aktor mendapatkan informasi detail simpanan.

**Table 4.31 Skenario Use Case Melihat Detail Pinjaman (Tambahan dari Fase Elaborasi)**

<i>Flow of Events</i> untuk <i>use case</i> melakukan melihat detail pinjaman	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat mendapatkan detail dari pinjaman yang dimiliki
<i>Actor</i>	Anggota
<i>Pre-conditions</i>	1. Aktor berada pada halaman pinjaman
<i>Basic flow of Events</i>	1. Aktor memilih fungsi lihat detail pinjaman 2. Sistem menampilkan data detail pinjaman yang dimiliki aktor
<i>Alternative Flows</i>	-
<i>Post-conditions</i>	Aktor mendapat informasi detail pinjaman yang dimiliki

Table 4.31 menjelaskan skenario bagaimana aktor dapat mendapatkan detail dari pinjaman yang dimiliki. Aktor yang berperan dalam *use case* ini adalah

anggota. *Use case* ini dapat dilakukan ketika aktor telah masuk kedalam sistem dan berada pada halaman pinjaman. Akhir dari *use case* ini adalah aktor mendapatkan informasi detail pinjaman.

**Table 4.32 Skenario *Use Case* Menghitung Estimasi Pinjaman (Tambahan dari Fase Elaborasi)**

<i>Flow of Events</i> untuk <i>use case</i> melakukan perhitungan estimasi pinjaman	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat mendapatkan estimasi pinjaman
<i>Actor</i>	Anggota
<i>Pre-conditions</i>	1. Aktor berada pada halaman pengajuan pinjaman
<i>Basic flow of Events</i>	1. Aktor menginputkan besar pinjaman dan lama pinjaman 2. Sistem menampilkan hasil perhitungan dari pengajuan yang diinputkan
<i>Alternative Flows</i>	A1 Input tidak sesuai format  Pada tahapan sistem menampilkan hasil perhitungan, jika nilai pinjaman bernilai minus atau kosong maka sistem akan menampilkan notifikasi data yang dimasukkan tidak valid.
<i>Post-conditions</i>	Aktor mendapat informasi estimasi pinjaman

Table 4.32 menjelaskan skenario bagaimana aktor dapat mengetahui estimasi pinjaman. Aktor yang berperan dalam *use case* ini adalah anggota. *Use case* ini dapat dilakukan ketika aktor berada pada halaman pengajuan pinjaman. Terdapat *alternative flows* ketika data yang dimasukkan salah maka sistem akan menampilkan notifikasi kesalahan. Akhir dari *use case* ini adalah aktor mengetahui estimasi pinjaman yang akan diajukan.

**Table 4.33 Skenario *Use Case* Unduh Form Pengajuan (Tambahan dari Fase Elaborasi)**

<i>Flow of Events</i> untuk <i>use case</i> melakukan unduh form pengajuan	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat mengunduh file dari sistem
<i>Actor</i>	Anggota
<i>Pre-conditions</i>	1. Aktor telah masuk kedalam sistem
<i>Basic flow of Events</i>	1. Aktor menekan tombol unduh <i>form</i> 2. Sistem akan melakukan pengambilan <i>file</i>
<i>Alternative Flows</i>	-

<i>Post-conditions</i>	Aktor berhasil mengunduh file
------------------------	-------------------------------

Table 4.33 menjelaskan skenario bagaimana aktor dapat mengunduh *form* pengajuan pinjaman. Aktor yang berperan dalam *use case* ini adalah anggota. *Use case* ini dapat dilakukan ketika berada pada halaman pengajuan. Akhir dari *use case* ini adalah aktor

**Table 4.34 Skenario *Use Case* Edit *Password* (Tambahan dari Fase Elaborasi)**

<i>Flow of Events</i> untuk <i>use case</i> melakukan edit <i>password</i>	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat merubah <i>password</i>
<i>Actor</i>	Anggota
<i>Pre-conditions</i>	1. Aktor telah masuk kedalam sistem
<i>Basic flow of Events</i>	1. Aktor memilih menu edit <i>password</i> 2. Sistem akan menampilkan <i>form</i> edit 3. Aktor meingputkan data 4. Sistem menyimpan <i>password</i> telah diubah
<i>Alternative Flows</i>	A1. Form tidak lengkap Pada tahap aktor menginputkan data, jika form tidak diisi maka akan menampilkan pesan field harus diisi A2. Password lama salah Pada tahapan aktor menginputkan data, jika pada data <i>password</i> lama tidak sesuai dengan <i>password</i> yang tersimpan dalam basis data maka sistem menampilkan pesan <i>password</i> lama tidak sesuai.
<i>Post-conditions</i>	Aktor memiliki <i>password</i> baru

Table 4.34 menjelaskan skenario bagaimana aktor dapat mengubah *password* yang dimiliki. Aktor yang berperan dalam *use case* ini adalah anggota. *Use case* ini dapat dilakukan ketika aktor telah masuk kedalam sistem. Terdapat *alternative flows* ketika *password* lama yang dimasukkan tidak sesuai dengan *password* yang tersimpan dalam basis data. Akhir dari *use case* ini adalah aktor *password* anggota dapat berubah.

**Table 4.35 Skenario *Use Case* Melihat Detail Pengajuan (Tambahan dari Fase Konstruksi)**

<i>Flow of Events</i> untuk <i>use case</i> melihat detail pengajuan	
<i>Brief Description</i>	<i>Use case</i> ini menggambarkan bagaimana aktor dapat merubah melihat detail pengajuan
<i>Actor</i>	Anggota

<i>Pre-conditions</i>	<ol style="list-style-type: none"> <li>1. Aktor memiliki pengajuan pinjaman</li> <li>2. Aktor berada pada halaman pengajuan pinjaman</li> </ol>
<i>Basic flow of Events</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu detail pengajuan</li> <li>2. Sistem akan detail pengajuan yang telah dibuat</li> </ol>
<i>Alternative Flows</i>	-
<i>Post-conditions</i>	Aktor memiliki dapat melihat detail pengajuan yang telah dilakukan

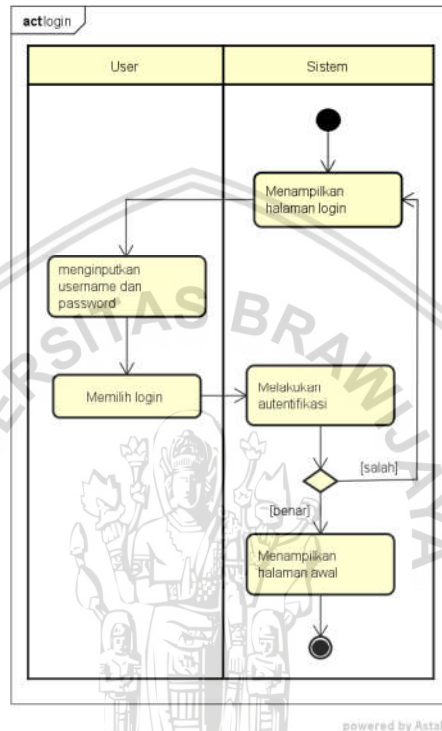
Table 4.35 menjelaskan skenario bagaimana aktor dapat mengubah melihat detail pengajuan yang telah dilakukan. Aktor yang berperan dalam *use ase* ini adalah anggota. *Use case* ini dapat dilakukan ketika aktor mempunyai pengajuan pinjaman yang sedang diproses oleh petugas. Akhir dari *use case* ini adalah aktor dapat mengetahui detail dari pengajuan yang telah dilakukan dan dalam tahap proses persetujuan.



#### 4.3.2 Pemodelan Activity Diagram

Perancangan *activity diagram* dibuat berdasar pada skenario *use case* yang telah dibuat. *Activity diagram* bertujuan untuk menggambarkan alur dari sebuah *use case*. Berikut merupakan *activity diagram* yang ada pada sistem informasi *customer touching applications* Unti Simpan Pinjam KPRI

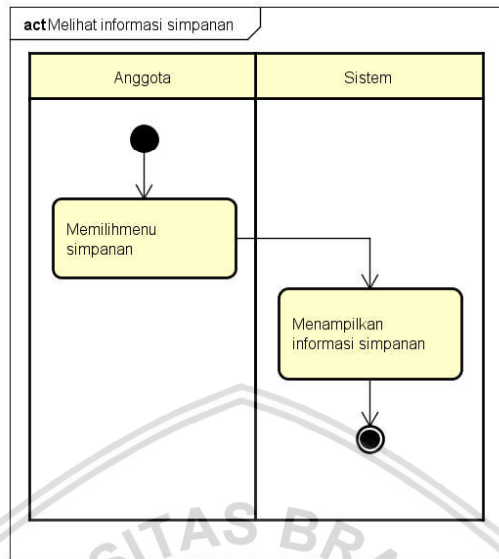
##### 1. Activity Diagram Login



Gambar 4.13 Activity diagram login

Gambar 4.13 merupakan *activity diagram login*, aktifitas dimulai ketika aktor mengakses halaman *login* yang ada pada sistem dimana terdapat form *username* dan *password*. Setelah sistem menampilkan halaman *login* langkah selanjutnya adalah aktor memasukkan *username* dan *password* yang akan dilakukan autentifikasi oleh sistem. Jika hasil autentifikasi benar maka sistem akan menampilkan halaman awal tetapi jika autentifikasi salah maka sistem akan menampilkan halaman *login* kembali.

## 2. Activity Diagram Melihat Informasi Simpanan

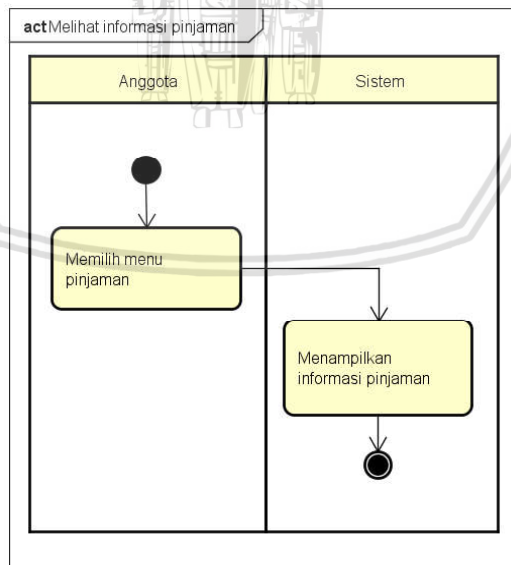


powered by Astah

**Gambar 4.14 Activity diagram melihat info simpanan**

Gambar 4.14 merupakan *activity diagram* melihat informasi simpanan yang menggambarkan bagaimana aktifitas untuk melihat informasi simpanan pada sistem. Aktifitas dimulai ketika aktor memilih menu melihat simpanan maka sistem akan menampilkan informasi simpanan yang dimiliki oleh aktor.

## 3. Activity Diagram Melihat Informasi Pinjaman



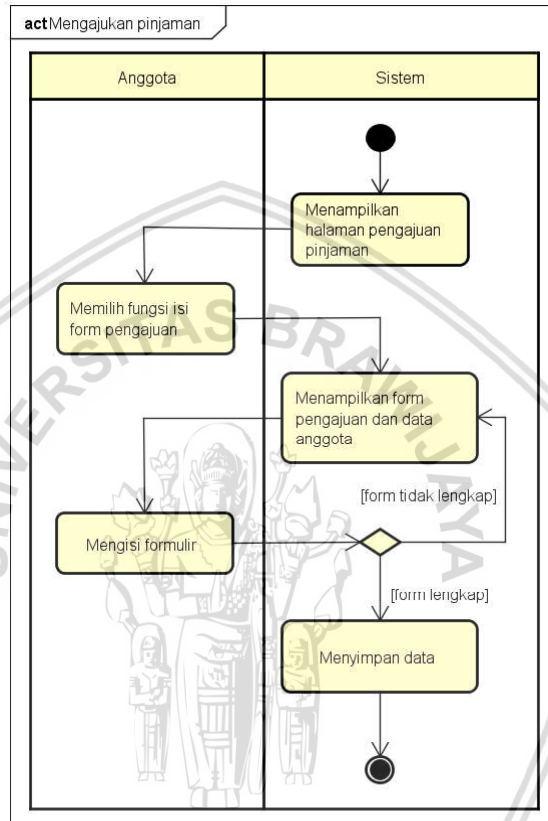
powered by Astah

**Gambar 4.15 Activity diagram melihat informasi pinjaman**



Gambar 4.15 merupakan *activity diagram* melihat informasi pinjaman yang menggambarkan bagaimana aktivitas untuk melihat informasi pinjaman pada sistem. Aktivitas dimulai ketika aktor memilih menu pinjaman maka sistem akan menampilkan informasi pinjaman yang dimiliki oleh aktor.

#### 4. Activity Diagram Pengajuan Pinjaman

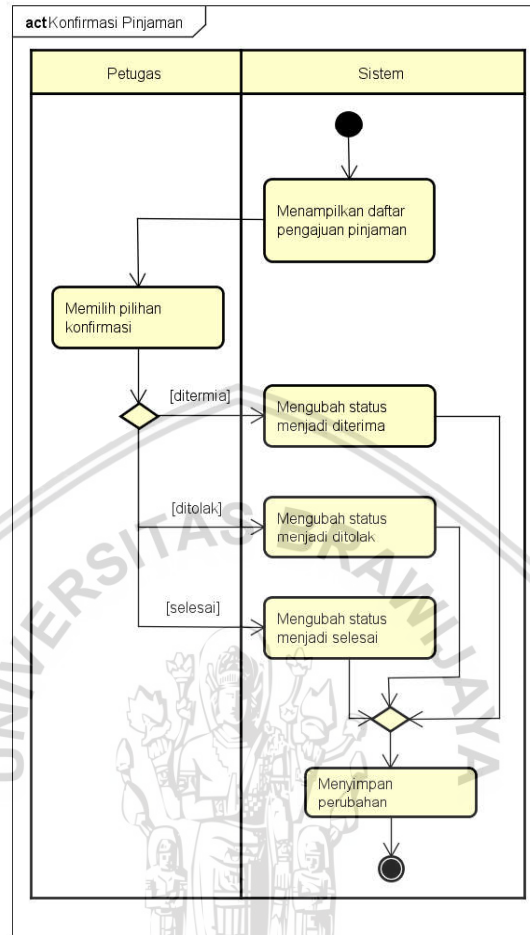


powered by Astah

**Gambar 4.16 Activity Diagram pengajuan pinjaman**

Gambar 4.16 merupakan *activity diagram* pengajuan pinjaman yang menggambarkan bagaimana aktivitas untuk mengajukan pinjaman. Aktivitas dimulai ketika aktor memilih fungsi isi form dan berakhir ketika sistem menyimpan data dari formulir pendaftaran.

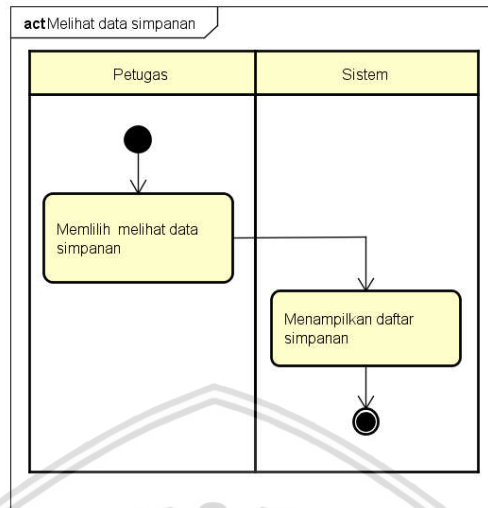
## 5. Activity Diagram Konfirmasi Pinjaman



Gambar 4.17 Activity Diagram konfirmasi pinjaman

Gambar 4.17 merupakan *activity diagram* konfirmasi pinjaman yang menggambarkan aktivitas untuk melakukan konfirmasi terhadap permohonan pinjaman yang ada. Aktivitas dimulai ketika sistem menampilkan daftar permohonan pinjaman dan aktor memilih jenis konfirmasi yang nantinya akan disimpan sistem.

## 6. Activity Diagram Melihat Data Simpanan

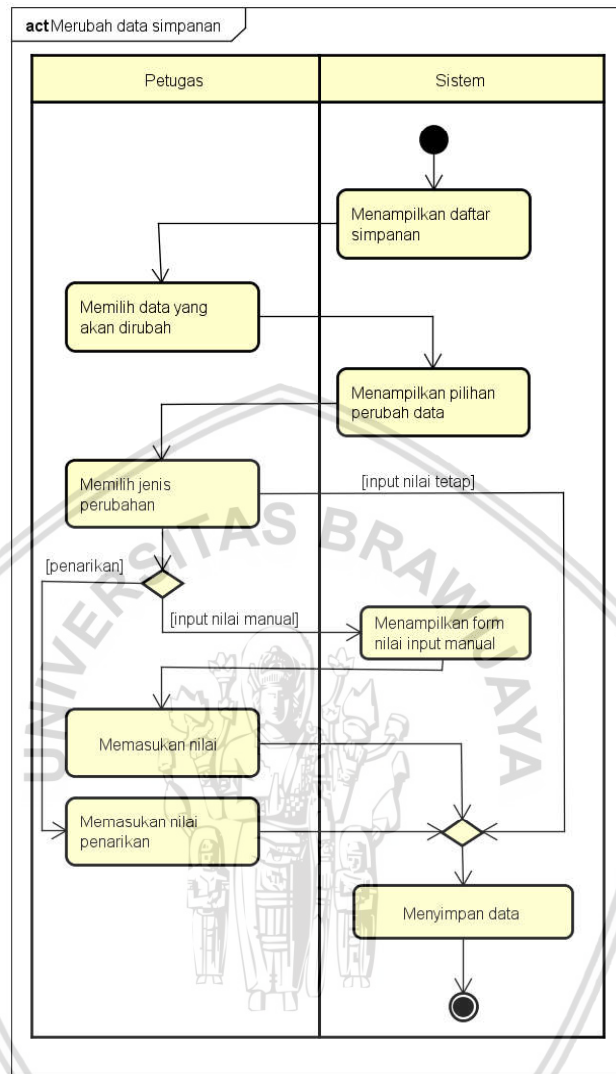


powered by Astah

**Gambar 4.18 Activity diagram melihat data simpanan**

Gambar 4.18 merupakan *activity diagram* melihat data simpanan yang menggambarkan aktivitas petugas untuk melihat seluruh data simpanan. Aktivitas dimulai ketika aktor memilih menu simpanan. Sistem akan menampilkan daftar simpanan yang dimiliki oleh anggota unit simpanan pinjam. Aktivitas berakhir dengan ditampilkannya data simpanan oleh sistem.

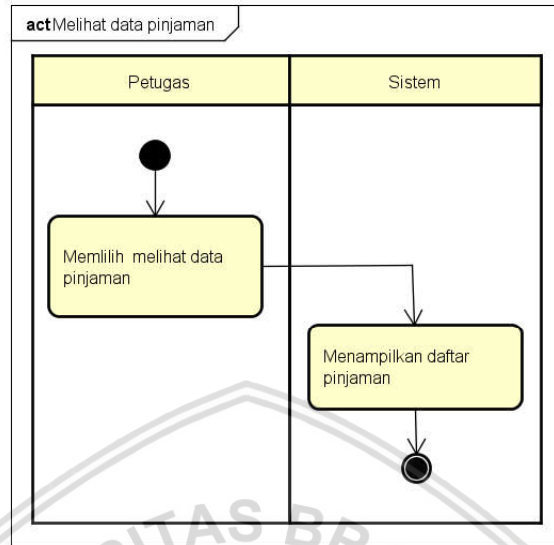
## 7. Activity Diagram Mengubah Data Simpanan



**Gambar 4.19 Activity diagram merubah data simpanan**

Gambar 4.19 merupakan *activity diagram* merubah data simpanan yang menggambarkan aktivitas untuk merubah data simpanan yang dimiliki oleh masing – masing anggota. Aktivitas dimulai menampilkan daftar simpanan anggota. Aktor memilih fungsi *update* untuk dapat melakukan perubahan pada data simpanan. Sistem akan menampilkan pilihan *update* data. Aktor memilih jenis *update* data untuk melakukan perubahan data.

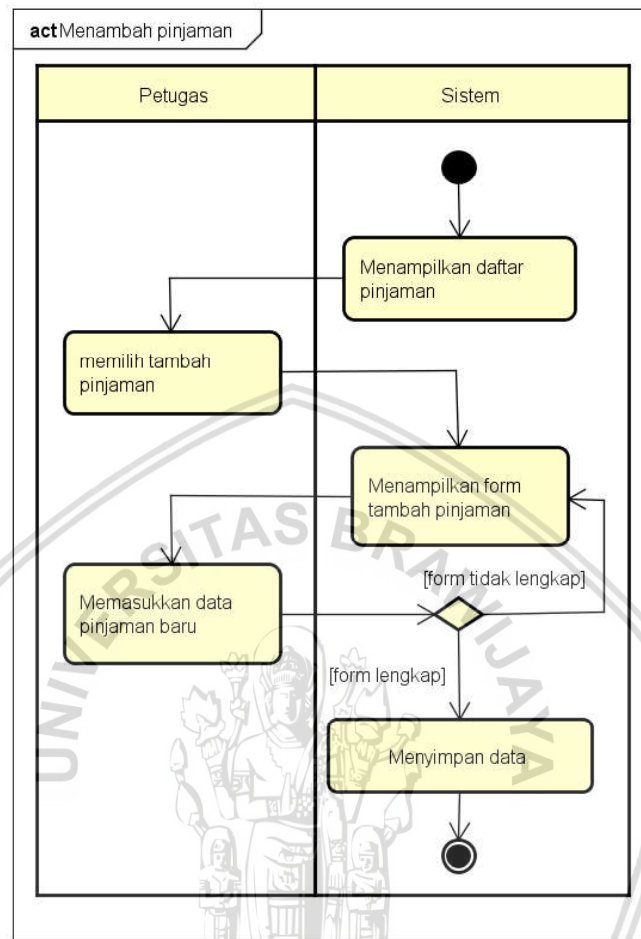
## 8. Activity Diagram Melihat Data Pinjaman



**Gambar 4.20 Activity diagram melihat data pinjaman**

Gambar 4.20 merupakan *activity diagram* melihat data pinjaman yang menggambarkan aktivitas untuk melihat data pinjaman yang dimiliki oleh masing – masing anggota. Aktivitas dimulai ketika aktor memilih fungsi pinjaman dan sistem akan menampilkan daftar pinjaman yang ada.

## 9. Activity Diagram Menambah Data Pinjaman



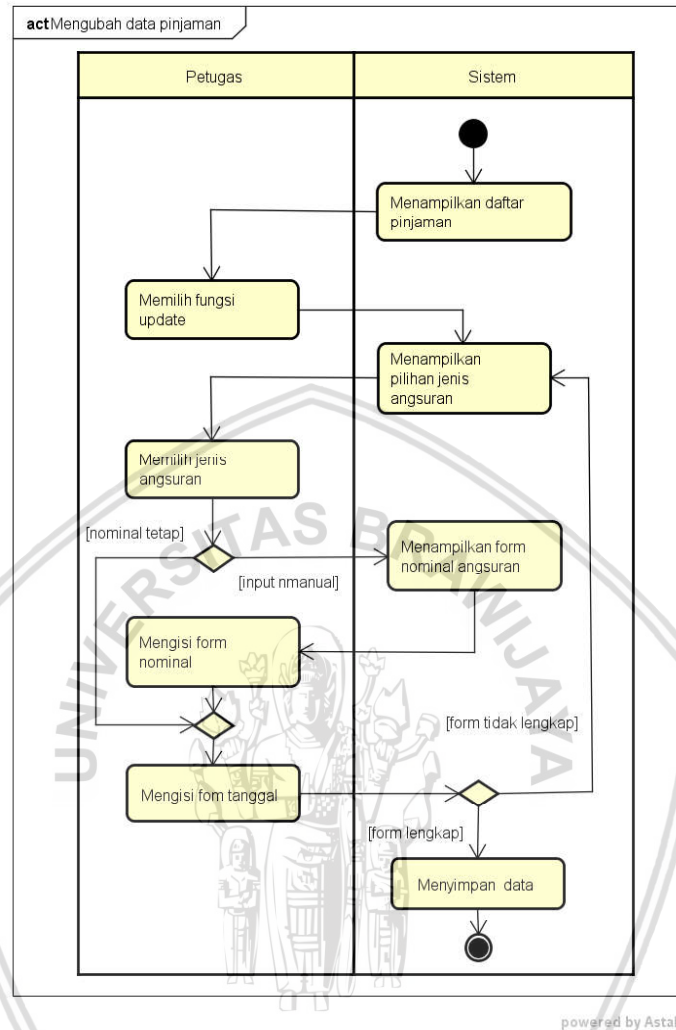
powered by Astah

**Gambar 4.21 Activity diagram tambah data pinjaman**

Gambar 4.21 merupakan *activity diagram* tambah data pinjaman yang menggambarkan aktivitas untuk menambah data pinjaman baru. Aktivitas dimulai ketika sistem menampilkan daftar pinjaman dan aktor memilih fungsi tambah pinjaman.



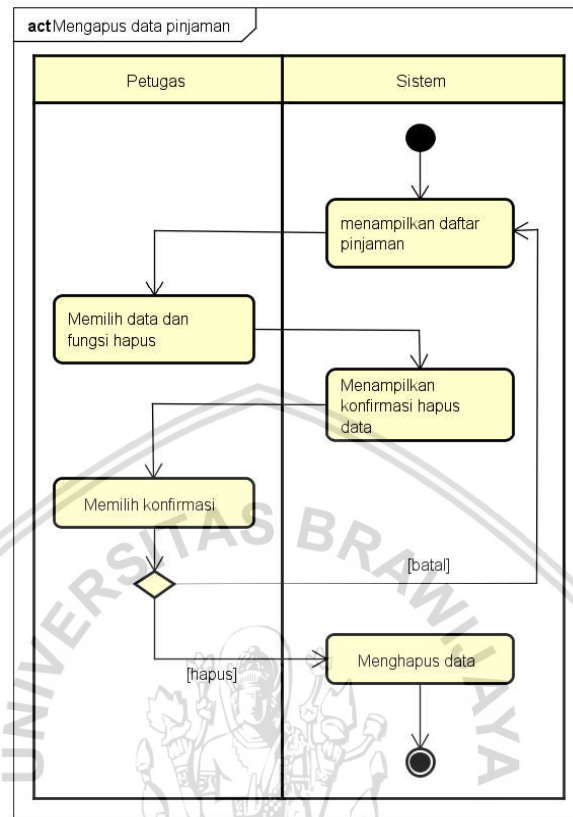
## 10. Activity Diagram Mengubah Data Pinjaman



Gambar 4.22 Activity diagram mengubah data pinjaman

Gambar 4.22 merupakan *activity diagram* mengubah data pinjaman yang menggambarkan aktivitas untuk mengubah data pinjaman. Aktivitas dimulai ketika sistem menampilkan daftar pinjaman dan aktor memilih fungsi update pinjaman. Sistem akan menampilkan jenis angsuran dan aktor akan memilih jenis angsuran. Aktivitas berakhir ketika sistem menyimpan data.

## 11. Activity Diagram Hapus Data Pinjaman

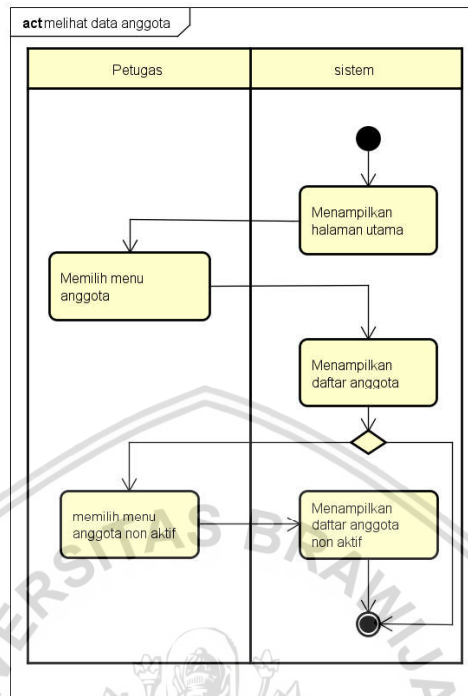


powered by Astah

**Gambar 4.23 Activity diagram hapus data pinjaman**

Gambar 4.23 merupakan *activity diagram* menghapus data pinjaman yang menggambarkan aktivitas untuk menghapus data pinjaman yang telah selesai. Aktivitas dimulai ketika sistem menampilkan daftar pinjaman dan aktor memilih fungsi hapus pinjaman. Sistem akan menampilkan konfirmasi hapus data dan ketika aktor memilih konfirmasi hapus maka sistem akan menghapus data pinjaman.

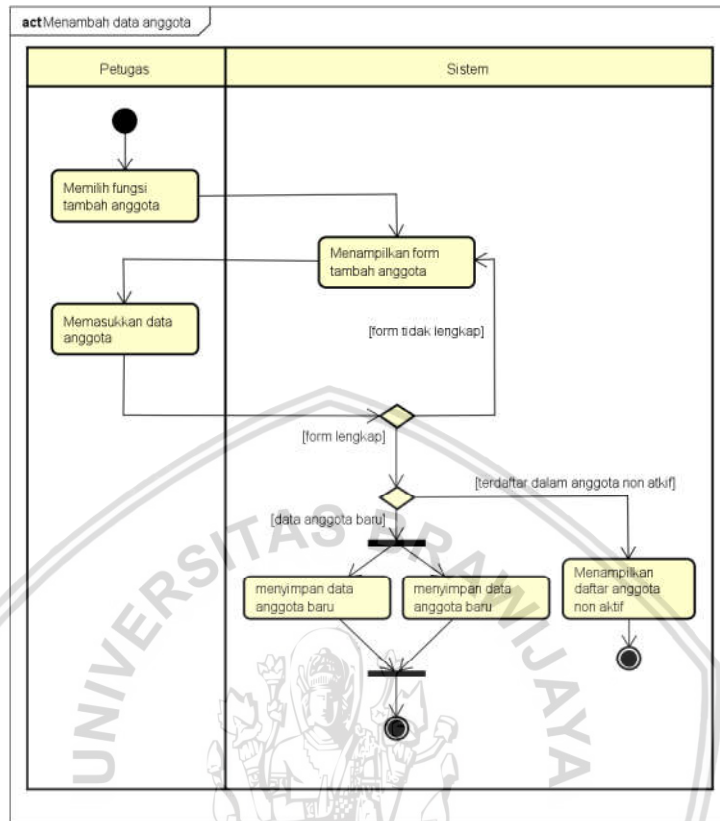
## 12. Activity Diagram Melihat Data Anggota



Gambar 4.24 ctivity diagram melihat data anggota

Gambar 4.24 merupakan *activity diagram* melihat data anggota yang menggambarkan aktivitas untuk melihat data anggota unit simpan pinjam. Aktor memilih menu anggota dan sistem akan menampilkan daftar anggota aktif. Aktor juga dapat memilih menu anggot non aktif untuk menampilkan daftar anggota yang tidak aktif.

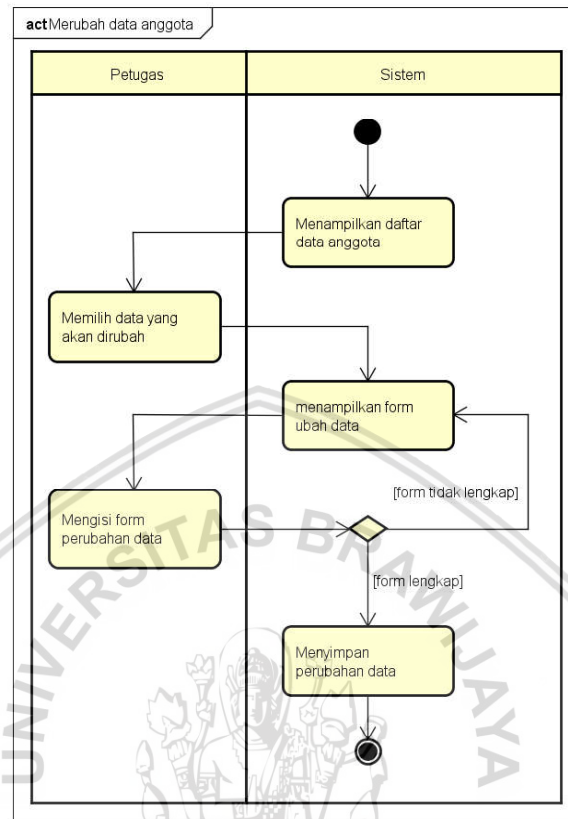
### 13. Activity Diagram Menambah Data Anggota



**Gambar 4.25 Activity diagram menambah data anggota**

Gambar 4.25 merupakan *activity diagram* menambah data anggota yang menggambarkan aktivitas untuk menambahkan data anggota baru pada unit simpan pinjam. Aktor memilih fungsi tambah anggota dan sistem akan menampilkan form tambah anggota yang dapat diisi dengan data anggota baru oleh aktor. Jika data yang dimasukkan sama dengan data anggota non aktif maka sistem akan menampilkan daftar anggota non aktif namun jika data yang dimasukkan merupakan data baru maka sistem akan menyimpan data tersebut.

## 14. Activity Diagram Merubah Data Anggota

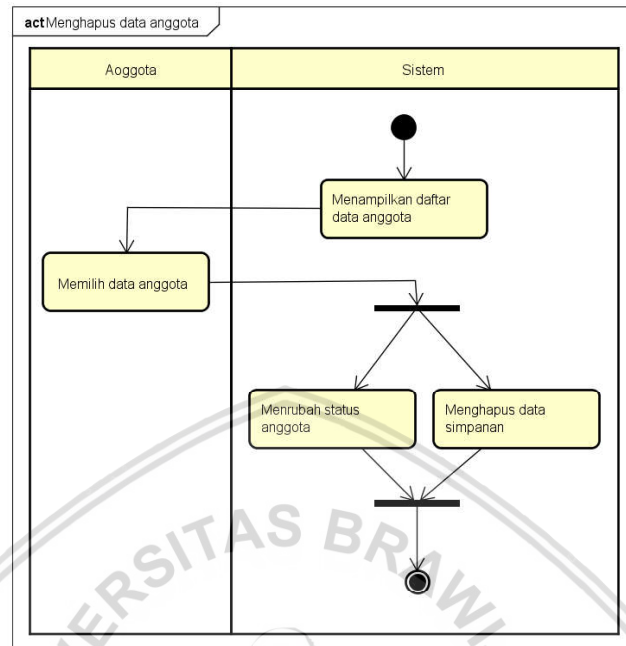


powered by Astah

Gambar 4.26 Activity diagram mengubah data anggota

Gambar 4.26 merupakan *activity diagram* mengubah data anggota yang menggambarkan aktivitas untuk melakukan perubahan terhadap data anggota unit simpan pinjam. Sistem akan menampilkan daftar anggota dan aktor dapat memilih data anggota yang akan diubah dengan memilih fungsi *update*. Setelah aktor mengisi form perubahan data sistem akan menyimpan data jika seluruh field form terisi.

## 15. Activity Diagram Menghapus Data Anggota

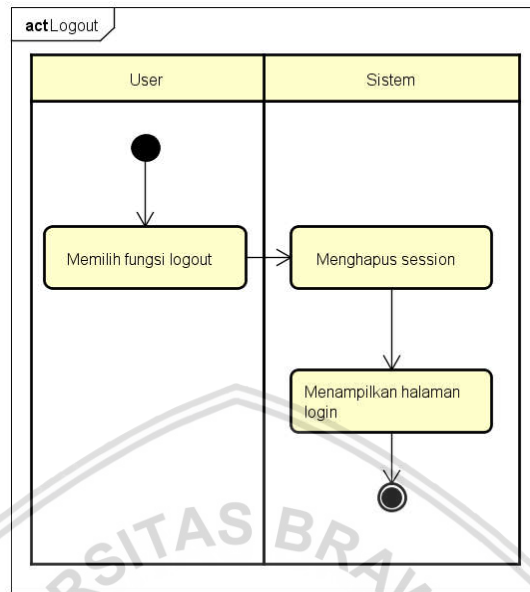


**Gambar 4.27 Activity diagram menghapus data anggota**

Gambar 4.27 merupakan *activity diagram* menghapus data anggota yang menggambarkan aktivitas untuk menghapus anggota dari unit simpan pinjam. Sistem akan menampilkan daftar anggota dan aktor dapat memilih data anggota yang akan dihapus. Sistem akan merubah status anggota menjadi non aktif dan menghapus data simpanan dari anggota tersebut.



## 16. Activity Diagram Logout

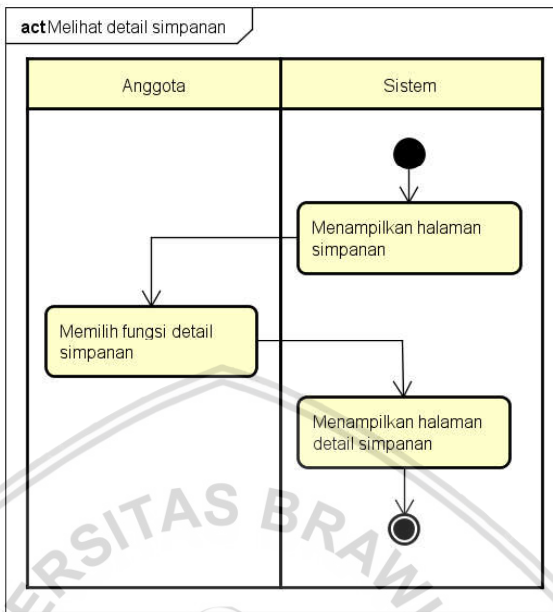


powered by Astah

**Gambar 4.28 Activity diagram logout**

Gambar 4.28 merupakan *activity diagram logout* yang menggambarkan aktivitas untuk keluar dari sistem. Aktor akan memilih fungsi *logout* sehingga sistem akan menghapus *session* dan menampilkan halaman login kembali.

## 17. Activity Diagram Melihat Detail Simpanan (tambahan pada fase elaborasi)

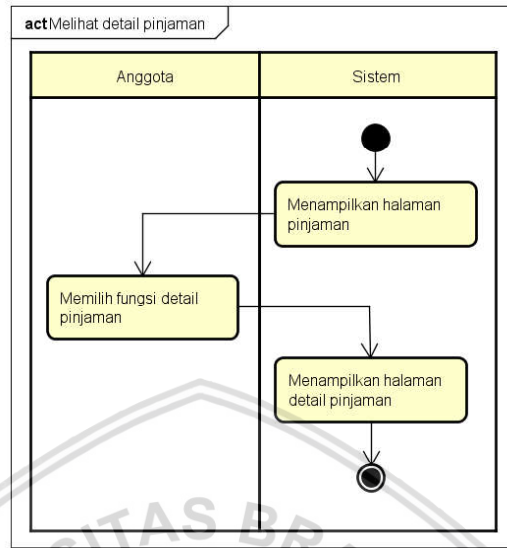


powered by Astah

**Gambar 4.29 Activity diagram melihat detail simpanan**

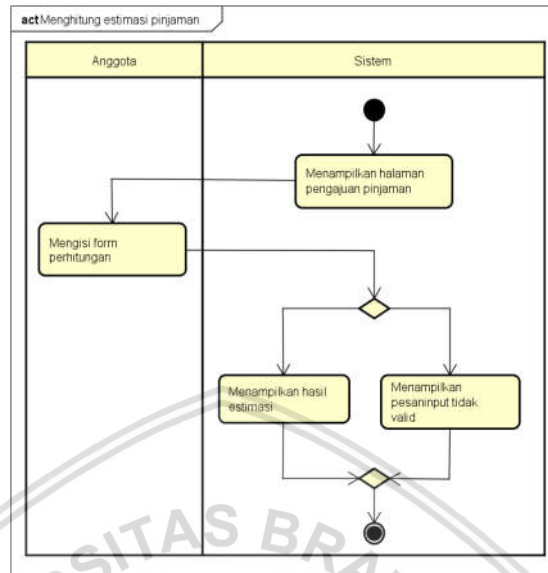
Gambar 4.29 merupakan *activity diagram* melihat detail simpanan menjelaskan bagaimana aktivitas untuk melihat detail dari setiap perubahan simpanan baik itu penambahan atau pengurangan. Aktivitas dimulai ketika sistem menampilkan halaman informasi simpanan. Ketika aktor memilih menu detail simpan maka sistem akan menampilkan detail pada simpanan yang dimiliki oleh aktor.

## 18. Activity Diagram Melihat Detail Pinjaman (tambahan pada fase elaborasi)

**Gambar 4.30 Activity diagram melihat detail pinjaman**

Gambar 4.30 merupakan *activity diagram* melihat detail pinjaman menjelaskan bagaimana aktifitas untuk melihat detail data perubahan pinjaman yaitu angsuran pinjaman. Aktivitas dimulai ketika sistem menampilkan halaman informasi pinjaman. Ketika aktor memilih menu detail pinjaman maka sistem akan menampilkan detail pada pinjaman yang dimiliki oleh aktor.

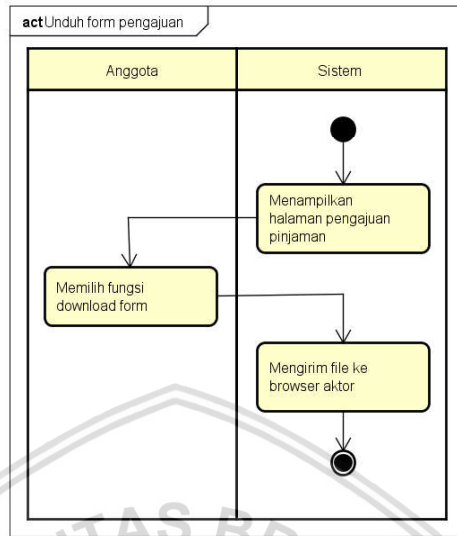
## 19. Activity Diagram Menghitung Estimasi (tambahan dari fase elaborasi)



Gambar 4.31 Activity diagram menghitung estimasi

Gambar 4.31 merupakan *activity diagram* menghitung estimasi pinjaman yang menggambarkan bagaimana aktor dapat mendapatkan estimasi angsuran dan bunga dari pinjaman yang akan diajukan. Aktivitas dimulai ketika aktor mengakses halaman pengajuan pinjaman. Pada bagian perhitungan estimasi aktor akan memasukkan estimasi pinjaman. Sistem akan melakukan perhitungan estimasi dan menampilkan hasil estimasi tersebut.

## 20. Activity Diagram Unduh File (tambahan dari fase elaborasi)

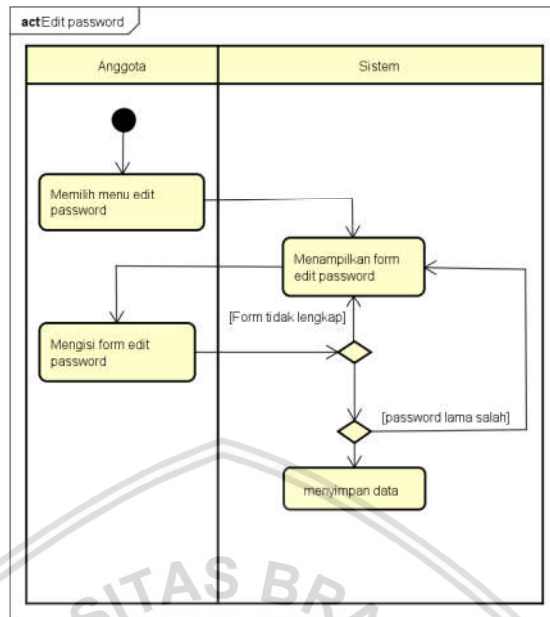


**Gambar 4.32 Activity diagram unduh file**

Gambar 4.32 merupakan *activity diagram* unduh file dimulai ketika aktor mengakses halaman pengajuan pinjaman dalam sistem. Aktor kemudian memilih unduh file untuk mendapatkan file dari sistem. Kemudian sistem akan memberikan file kepada aktor.

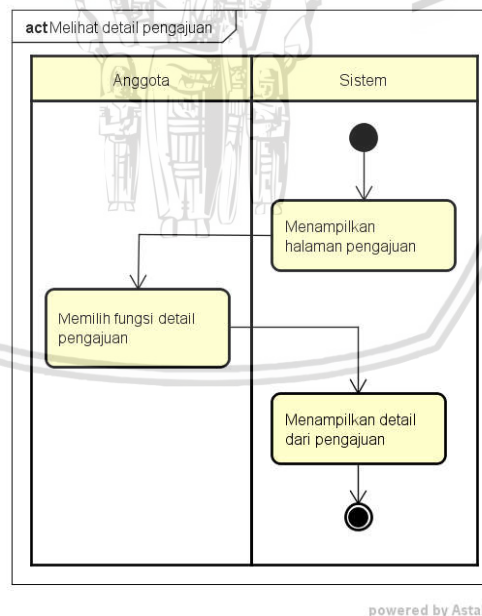
## 21. Activity Diagram Edit Password (tambahan dari fase elaborasi)

Gambar 4.33 merupakan *activity diagram* mengganti *password*, dimulai ketika aktor mengakses halaman edit *password* dalam sistem. Aktor kemudian menginputkan *password* lama dan *password* baru. Sistem akan melakukan pengecekan terhadap *password* lama dan akan menyimpan perubahan ketika pengecekan berhasil. Sistem akan menampilkan pesan hasil dari edit *password*.



**Gambar 4.33 Activity diagram edit password**

## 22. Activity Diagram Melihat Detail Pengajuan (tambahan dari fase konstruksi)



**Gambar 4.34 Activity diagram melihat detail pengajuan**

Gambar 4.35 merupakan *activity diagram* melihat detail pengajuan yang telah dilakukan oleh anggota. Aktivitas dimulai ketika sistem menampilkan halaman pengajuan dan aktor memilih fungsi detail



pengajuan. Sistem akan menampilkan detail dari pengajuan yang telah dibuat oleh anggota.



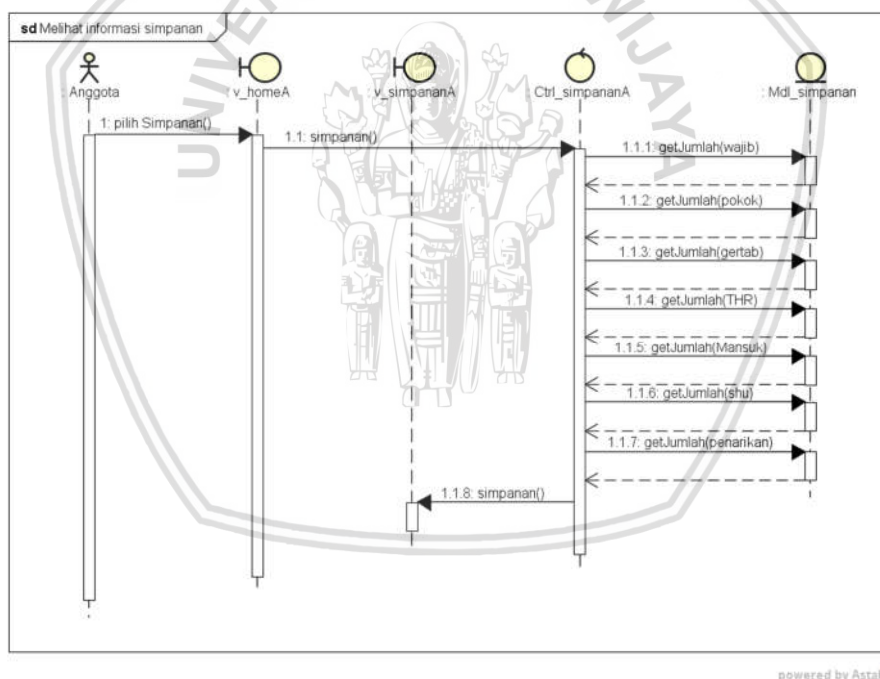
## BAB 5 PERANCANGAN

Pada bab ini berisi perancangan dari sistem yang akan dibangun, perancangan sistem ini terdiri atas perancangan *sequence diagram*, *design-level class diagram*, *physical data model*, *algoritma* dan perancangan antarmuka. Dalam metode RUP bab ini termasuk dalam fase *elaborasi* atau fase kedua dalam 4 langkah metode RUP.

### 5.1 Sequence Diagram

Perancangan *sequence diagram* dibuat berdasarkan pada masing – masing skenario *use case* yang telah didefinisikan pada analisis kebutuhan. Perancangan *sequence diagram* pada sistem informasi *customer touching applications* unit simpan pinjam KRPI “GURU” Kecamatan Sumbermanjing Wetan berfungsi untuk menggambarkan perilaku dalam sebuah skenario. Beberapa *sequence diagram* yang dirancang dalam penelitian ini antara lain :

#### 5.1.1 Sequence Diagram Melihat Informasi Simpanan



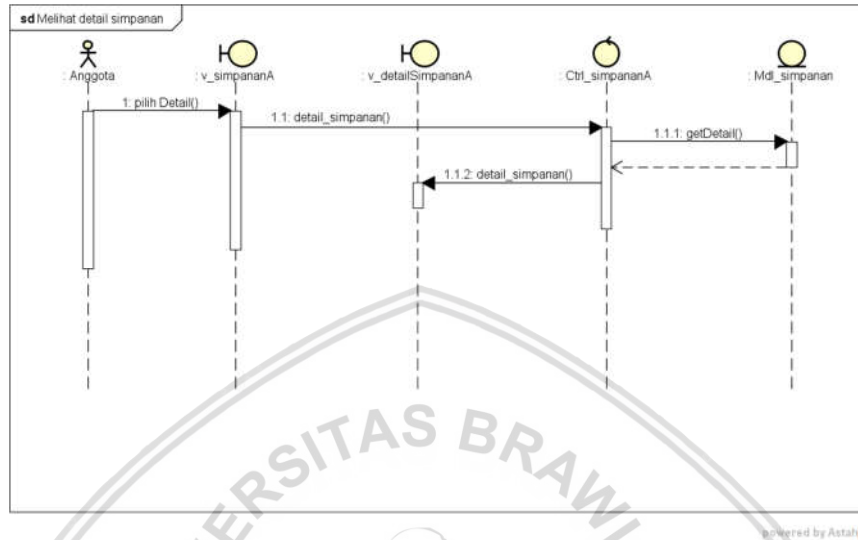
powered by Astah

**Gambar 5.1 Sequence Diagram Melihat Informasi Simpanan**

*Sequence diagram* pada gambar 5.1 menggambarkan interaksi antar objek dalam proses melihat informasi simpanan. Proses dimulai ketika anggota memilih menu simpanan yang akan menjalankan fungsi `simpanan()` pada `Ctrl_simpananA`. Kemudian sistem akan mengambil data simpanan menggunakan fungsi `getJumlah()` pada model `Mdl_simpanan`. Fungsi ini dijalankan beberapa kali sebanyak jenis dari simpanan yang ada. Setelah semua data telah didapatkan

sistem akan mengarahkan pada halaman v\_simpananA untuk menampilkan seluruh informasi simpanan anggota.

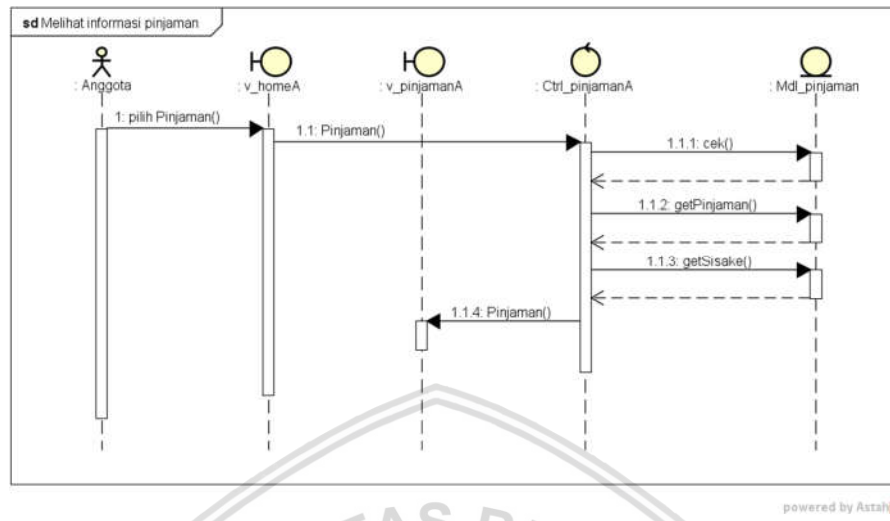
### 5.1.2 Sequence Diagram Melihat Detail Simpanan



**Gambar 5.2 Sequence Diagram Melihat Detail Simpanan**

*Sequence diagram* pada gambar 5.2 menggambarkan interaksi dalam proses melihat detail simpanan oleh anggota. Proses dimulai ketika anggota berada pada halaman v\_simpananA kemudian menjalankan fungsi detail\_simpanan() yang ada pada Ctrl\_simpananA. Sistem akan mengambil data dari basis data menggunakan fungsi getDetail() yang ada pada model Mdl\_simpanan. Setelah itu sistem akan mengarahkan pada halaman v\_detailSimpananA untuk menampilkan informasi detail simpanan anggota.

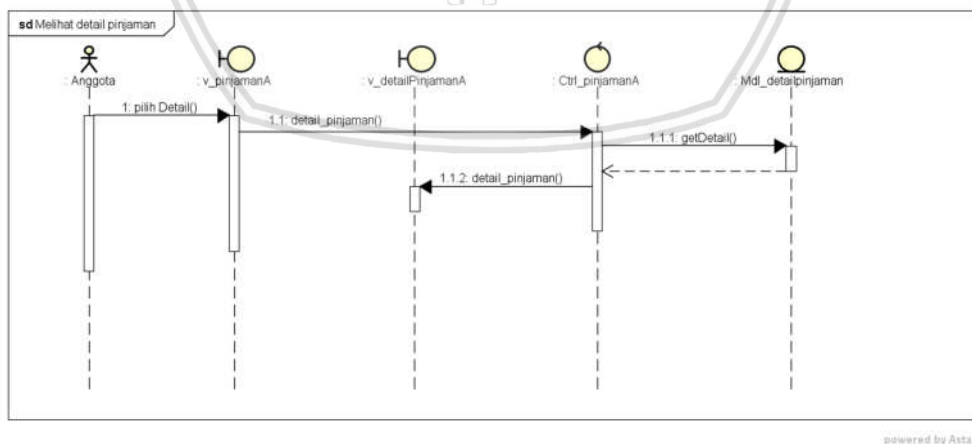
### 5.1.3 Sequence Diagram Melihat Informasi Pinjaman



Gambar 5.3 Sequence Diagram Melihat Informasi Pinjaman

Sequence diagram pada gambar 5.5 menggambarkan interaksi pada proses melihat informasi pinjaman oleh anggota. Proses dimulai ketika fungsi pinjaman() pada Ctrl\_pinjamanA dijalankan. Ctrl\_pinjamanA akan melakukan pengecekan ada tidaknya pinjaman yang dimiliki dengan fungsi cek() pada Mdl\_pinjaman. Setelah dilakukan pengecekan sistem akan mengambil data pinjaman dan sisa pinjaman dengan fungsi yang ada pada Mdl\_pinjaman. Controller Ctrl\_pinjamanA akan mengarahkan anggota ke v\_PinjamanA untuk menampilkan informasi pinjaman anggota.

### 5.1.4 Sequence Diagram Melihat Detail Pinjaman

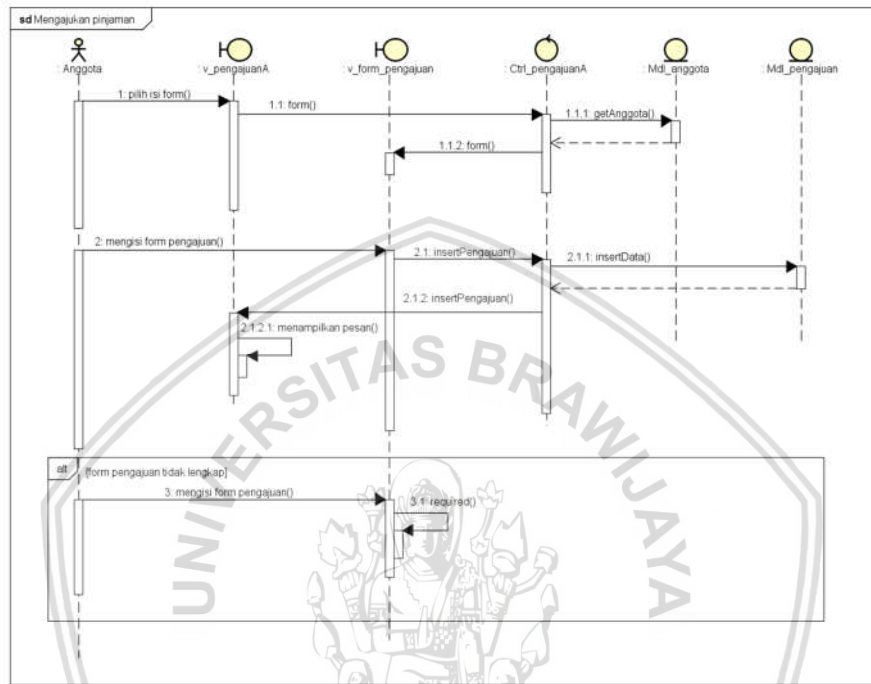


Gambar 5.4 Sequence Diagram Melihat Detail Pinjaman

Sequence diagram pada gambar 5.4 menggambarkan interaksi yang terjadi saat melihat detail pinjaman anggota. Proses dimulai ketika fungsi

detail\_pinjaman() pada Ctrl\_pinjamanA telah dijalankan. Sistem akan mengambil data pada basis data dengan bantuan fungsi getDetail() yang ada pada Mdl\_detailpinjaman. Setelah data diperoleh maka *controller* Ctrl\_pinjamanA akan mengarahkan pada v\_detailpinjaman.

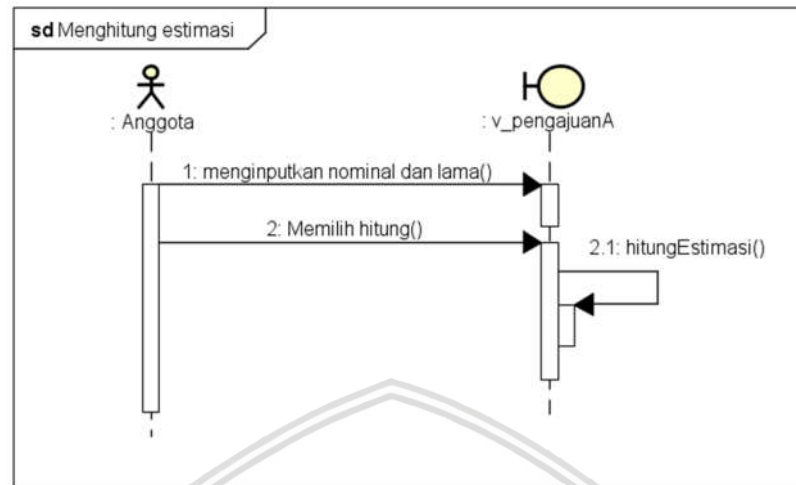
### 5.1.5 Sequence Diagram Mengajukan Pinjaman



**Gambar 5.5 Sequence Diagram Mengajukan Pinjaman**

*Sequence diagram* pada gambar 5.5 menggambarkan interaksi pada saat mengajukan pinjaman oleh anggota. Proses dimulai ketika anggota berada pada v\_pengajuanA dan menjalankan fungsi form() pada Ctrl\_pengajuanA yang kemudian mengambil data menggunakan fungsi getAnggota() pada Mdl\_anggota. Setelah itu *controller* Ctrl\_pengajuanA akan mengarahkan pada v\_form\_pengajuan dan menampilkan form untuk mengajukan pinjaman. Setelah anggota melakukan pengisian form, fungsi insertPengajuan() akan dijalankan sehingga mengaktifkan fungsi insertData() pada model Mdl\_pengajuan. Setelah fungsi insertData() dijalankan maka *controller* akan mengarahkan pada halaman v\_pengajuanA dan menampilkan pesan hasil dari pengajuan.

### 5.1.6 Sequence Diagram Menghitung Estimasi



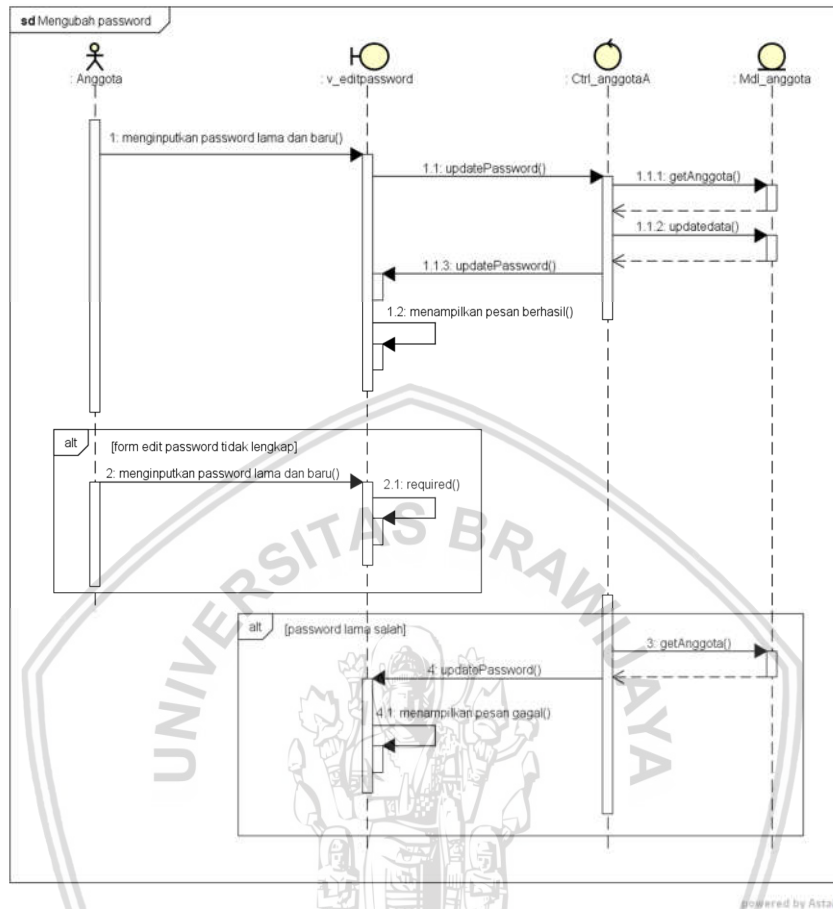
powered by Astah

**Gambar 5.6 Sequence Diagram Menghitung Estimasi**

*Sequence diagram* pada gambar 5.6 menggambarkan interaksi ketika melakukan perhitungan estimasi pinjaman oleh anggota. Proses dimulai ketika anggota menginputkan nilai dan lama kemudian memilih hitung yang menjalankan fungsi `hitungEstimasi()`. Dalam *sequence diagram* ini interaksi hanya ada pada 2 objek yaitu anggota dan `v_penajuanA` hal ini dikarenakan penghitungan estimasi tidak memerlukan data dari basis data dan proses penghitungan dapat dilakukan dengan menggunakan *jquery*.



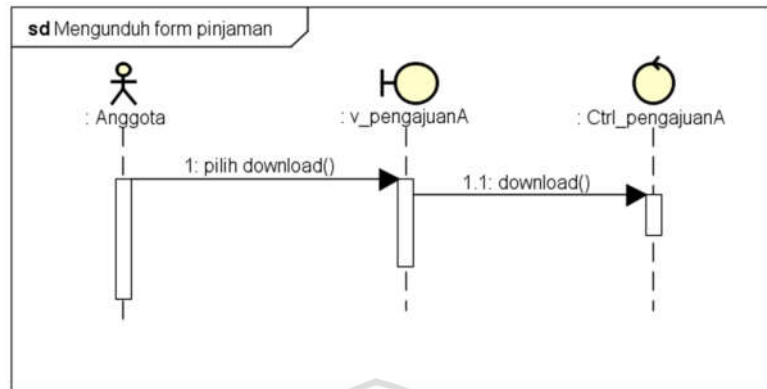
### 5.1.7 Sequence Diagram Mengubah Password



Gambar 5.7 Sequence Diagram Mengubah Password

Sequence diagram pada gambar 5.7 menggambarkan interaksi objek ketika proses mengubah *password* oleh anggota. Proses dimulai ketika fungsi `updatePassword()` pada `Ctrl_anggotaA` dijalankan. *Controller* `Ctrl_anggotaA` akan memanggil fungsi `getAnggota()` pada *model* `Mdl_anggota` untuk mendapatkan *password* yang saat ini ada pada basis data. Fungsi `updatedata()` digunakan untuk merubah data *password* yang tersimpan. Ketika *password* yang lama tidak sesuai dengan *password* yang di inputkan oleh anggota fungsi `updatedata()` tidak dijalankan.

### 5.1.8 Sequence Diagram Mengunduh Form Pinjaman

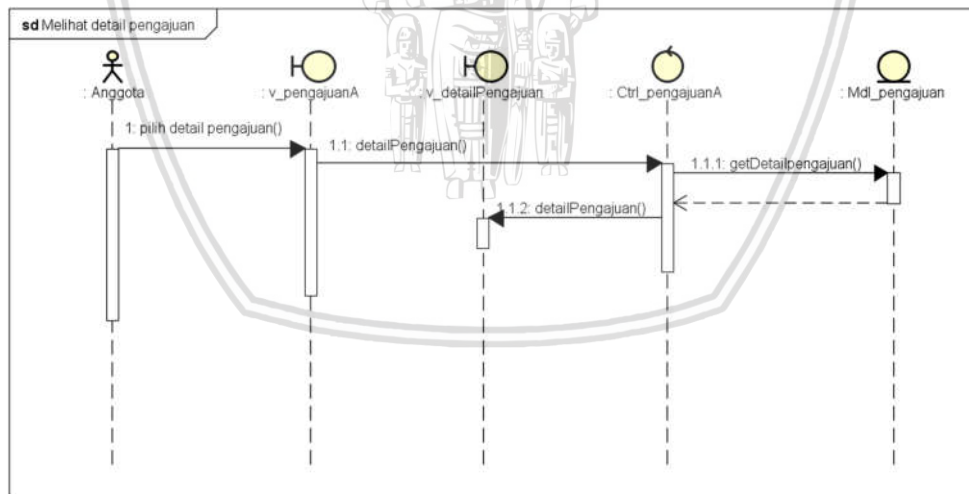


powered by Astah

**Gambar 5.8 Sequence Diagram Mengunduh Form Pinjaman**

*Sequence diagram* pada gambar 5.8 menggambarkan interaksi objek pada proses *download* form pinjaman. Proses dimulai ketika actor atau anggota memilih fungsi *download* yang ada pada *v\_pengajuanA* sehingga mengaktifkan *method download().method download()* yang berada pada *controller* *Ctrl\_pengajuanA* kemudian akan mengizinkan browser untuk mendapatkan file yang tersimpan dalam direktori sistem.

### 5.1.9 Sequence Diagram Melihat Detail Pengajuan



powered by Astah

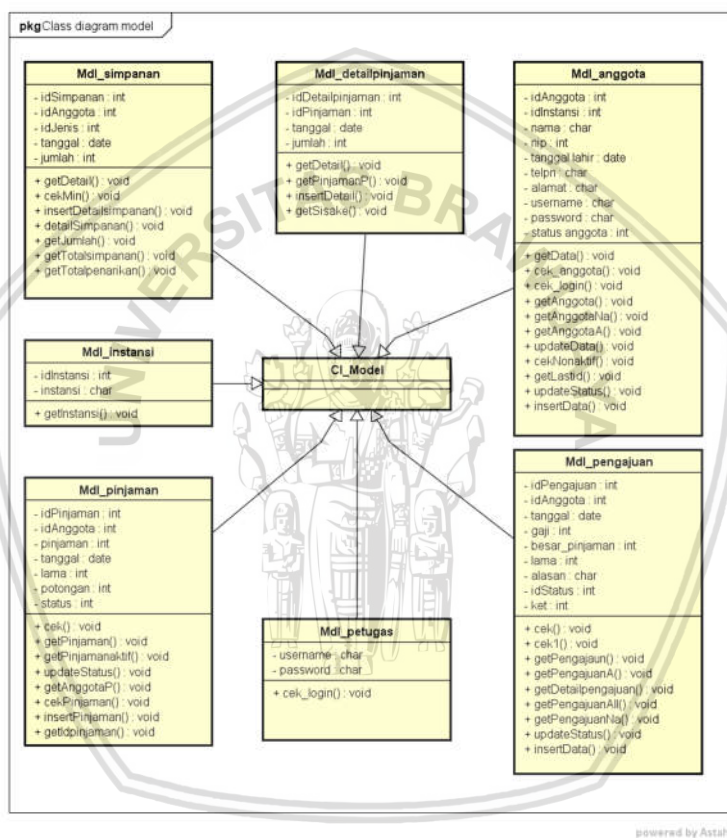
**Gambar 5.9 Sequence Diagram Melihat Detail Pengajuan**

*Sequence diagram* pada gambar 5.9 menggambarkan interaksi objek ketika proses melihat detail pengajuan oleh petugas. Proses dimulai ketika actor memilih detail pengajuan yang mengaktifkan fungsi *detailPengajuan()* pada *controller* *Ctrl\_pengajuanA*. Sistem akan mengambil data dengan fungsi

getDetailpengajuan() pada meodel Mdl\_pengajuan sehingga data dapat ditampilkan pada view v\_dataPengajuan.

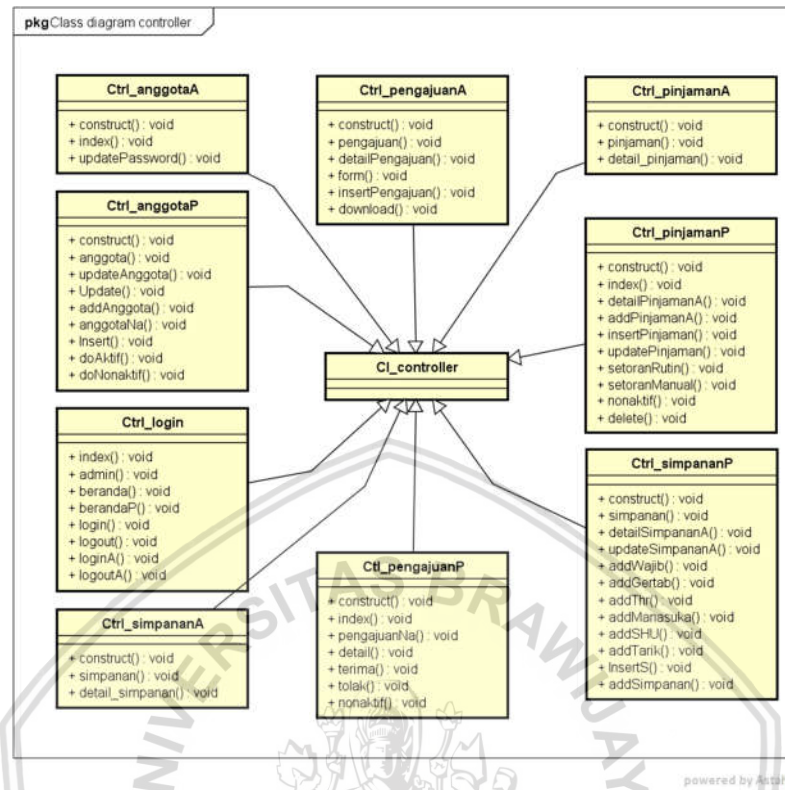
## 5.2 Design Class Diagram

Perancangan *class diagram* menggambarkan *class*, atribut, fungsi dan hubungan antar *class* satu dan lainnya. Perancangan dibuat berdasarkan objek – objek yang terdefiniskan pada perancangan *sequence diagram* dengan pada pola MVC yang ada pada *framework Codeigniter*. Berdasarkan pola MVC maka dalam perancangan *class diagram* sistem informasi *customer touching applicationss* terbagi menjadi *class diagram model* dan *class diagram controller*.



Gambar 5.10 Class Diagram Model

*Class diagram* model pada gambar 5.10 merupakan design class diagram model dalam pengembangan sistem informasi *costomer touching applications* KPRI, terdiri dari menjadi 7 model yang berfungsi untuk melakukan akses terhadap *database*. Terdiri dari 7 *class* yang mempunyai fungsi tersendiri, *class* Mdl\_anggota berfungsi untuk mengelola data anggota sedangkan Mdl\_petugas untuk mengelola data petugas. Mdl\_simpanan dan Mdl\_pinjaman berfungsi untuk mengelola atau mendapatkan data dari table simpanan dan pinjaman. Model Mdl\_detailpinjaman berfungsi untuk mengelola data dari table detail pinjaman sedangkan Mdl\_instansi digunakan untuk mendapatkan nilai dari table instansi. Kemudian terdapat Mdl\_pengajuan untuk mengelola data pengajuan pinjaman.



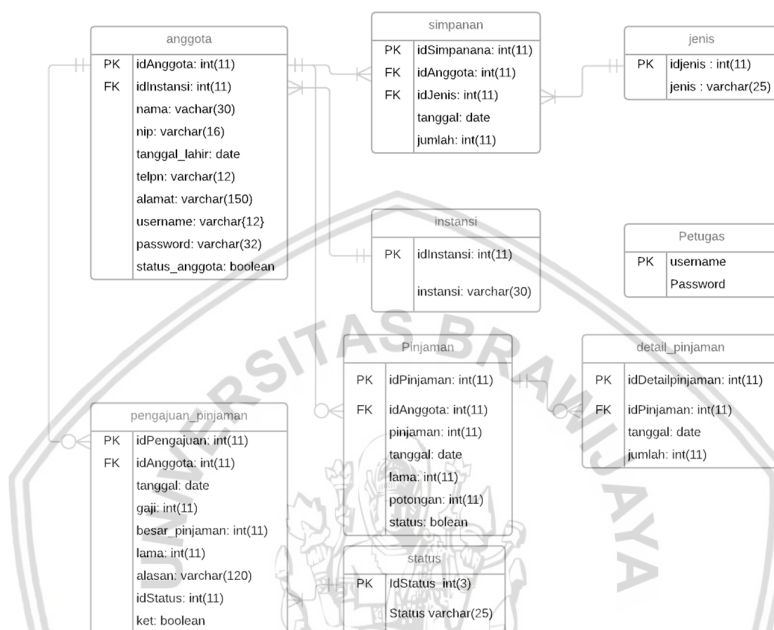
**Gambar 5.11 Class Diagram Controller**

*Class diagram controller* pada gambar 5.11 terbagi menjadi 9 class yang memiliki fungsi masing – masing. *Class Ctrl\_login* berfungsi untuk melakukan *login* bagi anggota maupun petugas dimana didalam *class* tersebut terdapat fungsi yang bertugas untuk memverifikasi proses *login*. *Class* selanjutnya merupakan *class Ctrl\_pinjamanA* dan *Ctrl\_simpananA* kedua kelas ini mempunyai fungsi yang hampir sama yaitu digunakan untuk mendapatkan informasi bagi anggota. *Class Ctrl\_pengajuanA* berfungsi untuk mengatur proses pengajuan pinjaman yang dilakukan oleh anggota sedangkan *Ctrl\_pengajuanP* berfungsi untuk mengatur proses konfirmasi oleh petugas. *Class Ctrl\_anggotaA* memfasilitasi proses edit *password* anggota sedangkan *class Ctrl\_anggotaP* untuk mengatur pengolahan data oleh petugas. Terakhir adalah *class Ctrl\_pinjamanP* dan *Ctrl\_simpananP* keduanya digunakan untuk mengelola informasi simpan pinjam oleh petugas.

### 5.3 Perancangan Basis Data

Perancangan basis data digunakan untuk membuat perancangan bagaimana relasi antar data dalam sistem. Dalam sistem informasi *customer touching application* unit simpan pinjam KPRI GURU Kecamatan Sumbermanjing Wetan perancangan basis data seperti pada gambar 5.12. Terdapat 9 table. Tabel anggota merupakan tabel untuk menyimpan data anggota meliputi nama, nip, alamat dan data penting lainnya sedangkan table petugas berisi data *username* dan *password* untuk petugas. Tabel simpanan merupakan table untuk menyimpan data simpanan yang dimiliki tiap anggota berkaitan dengan table jenis yang

menyimpan jenis – jenis dari simpanan yang ada pada unit simpan pinjam. Tabel pinjaman menyimpan data pinjaman yang dimiliki oleh anggota sedangkan tabel detail pinjaman berisikan data detail transaksi pembayaran pinjaman pada unit simpan pinjam. Tabel yang terakhir merupakan tabel pengajuan pinjaman yang menyimpan data pengajuan pinjaman yang dilakukan oleh anggota dan berrelasi dengan table status yang menyimpan status pengajuan pinjaman.



Gambar 5.12 Perancangan Basis Data

## 5.4 Perancangan Algoritma

Perancangan algoritma merupakan tahapan untuk menentukan serta menyusun sekumpulan operasi logika dari fungsi – fungsi yang telah didefinisikan pada *design class*. Halis dari perancangan algoritma akan dijadikan sebagai panduan dalam implementasi sistem. Berikut merupakan beberapa fungsi utama dari sistem informasi yang akan dikembangkan.

### 1. Perancangan Algoritma Melihat Informasi Simpanan

Table 5.1 Algoritma Fungsi Simpanan()

1	Mulai
2	Mengambil id anggota yang tersimpan dalam session
3	Menjalankan fungsi getJumlah dengan parameter id jenis simpanan tarik.
4	Inisiasi variable tarik dengan nilai total dari getJumlah tarik.
5	Menjalankan fungsi getJumlah dengan parameter id jenis simpanan pokok.
6	Inisiasi variable pokok dengan nilai total dari getJumlah pokok
7	Menjalankan fungsi getJumlah dengan parameter id jenis simpanan wajib.
8	



	Inisiasi variable wajib dengan nilai total dari getJumlah wajib
9	Menjalankan fungsi getJumlah dengan parameter id jenis simpanan gertab.
10	Inisiasi variable gertab dengan nilai total dari getJumlah gertab.
11	Menjalankan fungsi getJumlah dengan parameter id jenis simpanan thr.
12	Inisiasi variable thr dengan nilai total dari getJumlah thr.
13	Menjalankan fungsi getJumlah dengan parameter id jenis simpanan manasuka.
14	Inisiasi variable manasuka dengan nilai total dari getJumlah manasuka.
15	Menjalankan fungsi getJumlah dengan parameter id jenis simpanan shu.
16	Inisiasi variable shu dengan nilai total dari getJumlah shu.
17	Melakukan perhitungan dari seluruh variable
18	Deklarasi dan inisiasi array data
19	Memuat halaman simpanan
20	Selesai

## 2. Perancangan Algoritma Melihat Detail Simpanan

**Table 5.2 Algoritma Fungsi Detail\_simpanan()**

1	Mulai
2	Mengambil id yang tersimpan pada session.
3	Menjalankan fungsi getDetail untuk mendapat seluruh data simpanan
4	Deklarasi array data
5	Deklarasi variable hitungan dan total
6	Perulangan untuk mengeluarkan data dari fungsi getDetail
7	Jika bukan jenis penarikan
8	Variable total ditambahkan dengan jumlah
9	Menyimpan data kedalam array data
10	Lainnya
11	Variable total dikurangi dengan jumlah
12	Menyimpan data kedalam array data
13	Menampilkan halaman detail simpanan
14	Selesai

## 3. Perancangan Algoritma Mengajukan Pinjaman

**Table 5.3 Algoritma Fungsi InsertPengajuan()**

1	Mulai
2	Mengambil id anggota yang tersimpan dalam session
3	Deklarasi dan inisialisai array data yang diinputkan
4	Menjalankan fungsi insertdata sehingga data tersimpan dalam database
5	Menampilkan halaman pengajuan dan pesan data berhasil disimpan
6	Selesai

## 4. Perancangan Algoritma Melihat Detail Pengajuan

**Table 5.4 Algoritma Fungsi DetailPengajuan()**

1	Mulai
2	Mendeklarasikan variable data



3	Menjalankan fungsi <code>getDetailpengajuan</code> dengan parameter <code>id anggota</code>
4	Menampilkan halaman detail pengajuan
5	Selesai

## 5.5 Perancangan Antarmuka

Tahap perancangan antarmuka merupakan perancangan tampilan dari sistem yang akan dikembangkan dan termasuk dalam fase elaborasi pada metode pengembangan RUP. Perancangan antarmuka bertujuan untuk memberikan gambaran kepada pemangku kepentingan bagaimana tampilan yang akan menunjang interaksi sistem dan pengguna. Antarmuka dalam Sistem Informasi *Customer Touching Applications* Unit Simpan Pinjam KPRI GURU Kec Sumbermanjing Wetan memiliki tampilan anggota dan tampilan petugas. Berikut beberapa rancangan antarmuka sistem yang akan dikembangkan:

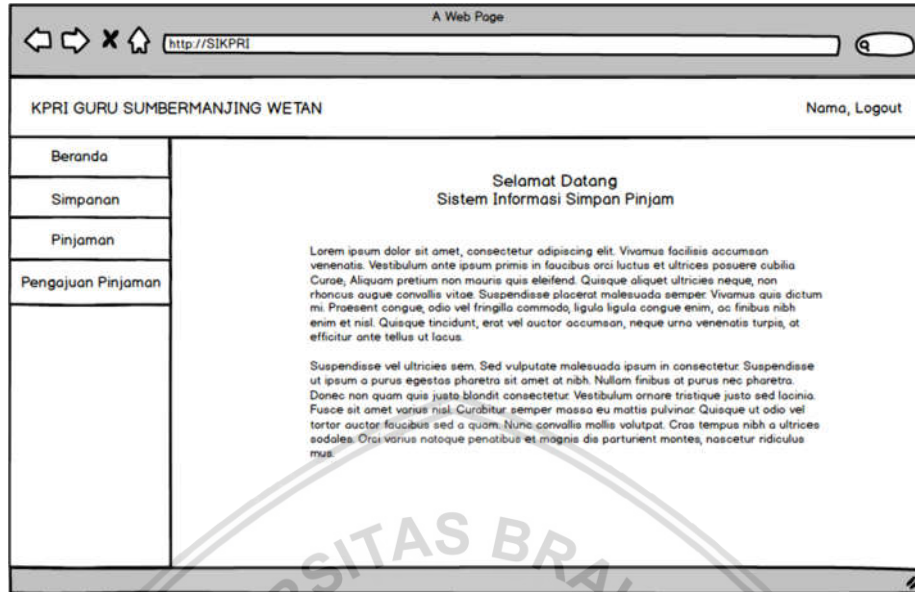
### 1. Perancangan Antarmuka *Login*

The image shows a web browser window with a login form. The browser's address bar displays 'http://SI[KPRI]'. The login form is titled 'Sistem Informasi Unit Simpan Pinjam KPRI GURU Sumbermanjing Wetan'. It contains two input fields: 'Username' and 'Password', each with a corresponding text box. Below these fields is a 'Login' button. The background of the browser window features a large, faint watermark of the Universitas Brawijaya logo.

**Gambar 5.13 Antarmuka *Login***

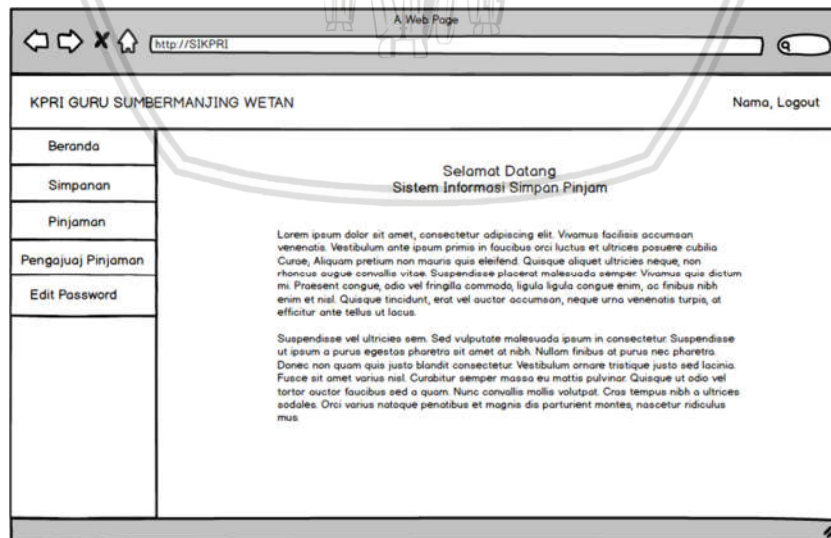
Gambar 5.13 merupakan rancangan antarmuka *login* menampilkan form berupa inputan *username* dan *password*. Selain itu juga terdapat tombol *login* untuk masuk kedalam sistem. Pada halaman *login* diberi judul sistem sehingga pengguna mengetahui bahwa halaman tersebut merupakan halaman *login* bagi sistem informasi unit simpan pinjam.

## 2. Perancangan Antarmuka *Home* Anggota



**Gambar 5.14 Antarmuka Beranda Anggota**

Halaman beranda anggota terdapat menu berupa beranda, simpanan, pinjaman dan pengajuan pinjaman. Pada pojok kanan atas juga terdapat informasi akun dan tombol *logout* bagi anggota. Sedangkan pada bagian utama terdapat deskripsi singkat tentang sistem tersebut. Rancang antar muka gambar 5.14 merupakan hasil rancangan pada awal fase elaborasi, sedangkan pada fase tersebut terdapat penambahan kebutuhan fungsional sistem seperti edit *password*. Sehingga hasil literasi fase elaborasi seperti gambar 5.15 dengan penambahan menu edit *password*.



**Gambar 5.15 Anatarmuka Beranda Anggota (literasi kedua fase elaborasi)**

### 3. Perancangan Antarmuka Simpanan

Pokok	Wajib	Getas	THR	Manasuka	Jumlah
-	-	-	-	-	-

**Gambar 5.16 Antarmuka Simpanan**

Gambar 5.17 merupakan rancangan antarmuka simpanan pada anggota menampilkan informasi mengenai simpanan yang dimiliki anggota berupa table. Informasi yang ditampilkan adalah besar simpanan berdasar pada jenis yang dimiliki oleh anggota. Pada literasi kedua fase elaborasi terdapat penambahan tombol detail yang berfungsi untuk menampilkan detail penarikan maupun setoran yang pernah dilakukan.

### 4. Perancangan Antarmuka Detail Simpanan (Tambahan Fase Elaborasi)

No	Tanggal	Pokok	Wajib	Getas	THR	Manasuka	Jumlah
-	-	-	-	-	-	-	-

No	Tanggal	Jumlah
-	-	-

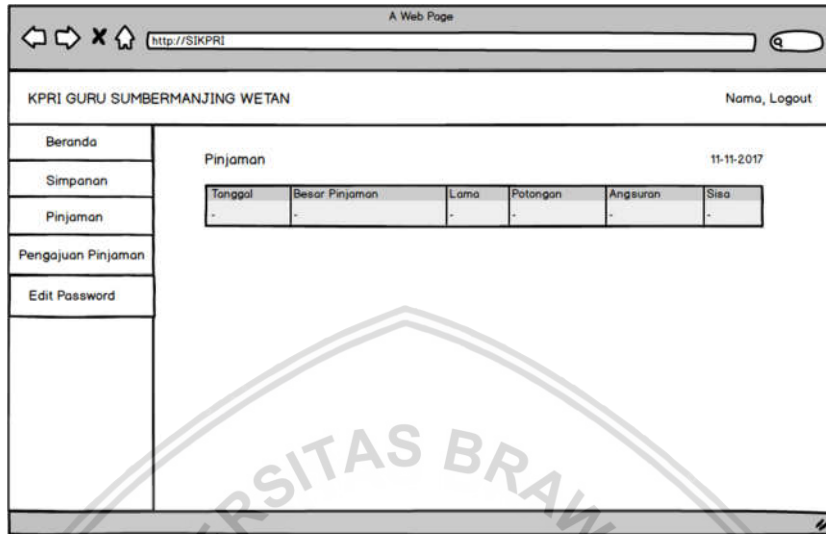
[Kembali](#)

**Gambar 5.17 Antarmuka Detail Simpanan**

Gambar 5.17 merupakan rancangan antarmuka halaman detail simpanan, menampilkan informasi berupa detail transaksi yang pernah dilakukan. Informasi yang ditampilkan meliputi detail setoran dan detail penarikan dari simpanan.

Informasi ditampilkan dalam bentuk tabel sehingga mudah untuk dipahami oleh pengguna.

##### 5. Perancangan Antarmuka Pinjaman



**Gambar 5.18 Antarmuka Pinjaman Anggota**

Gambar 5.18 merupakan rancangan antarmuka halaman pinjaman anggota untuk menampilkan informasi mengenai beban pinjaman yang dimiliki oleh anggota. Informasi tersebut ditampilkan dalam bentuk tabel sehingga informasi yang ditampilkan lebih mendetail. Pada literasi kedua fase elaborasi terdapat penambahan kebutuhan detail pinjaman sehingga terdapat penambahan tombol detail untuk menampilkan detail dari pinjamana berupa setoran pembayaran tiap bulannya.

##### 6. Perancangan Antarmuka Detail Pinjaman (Tambahan Fase Elaborasi)

**Gambar 5.19 Antarmuka Detail Pinjaman**

Gambar 5.19 merupakan rancangan antarmuka halaman detail pinjaman, menampilkan informasi berupa pembayaran yang telah dilakukan. Informasi ditampilkan dalam bentuk tabel sehingga mudah untuk dipahami oleh pengguna.

#### 7. Perancangan Antarmuka Pengajuan Pinjaman

**Gambar 5.20 Perancangan Antarmuka Pengajuan Pinjaman**

Gambar 5.20 merupakan rancangan antarmuka halaman pengajuan pinjaman terdapat informasi minimal dan maksimal pinjaman yang dapat diterima oleh anggota. Terdapat juga *form* untuk menghitung estimasi pinjaman yang berfungsi untuk menghitung estimasi potongan tiap bulan yang harus dibayarkan. Tombol buat permohonan akan menampilkan form untuk mengajukan permohonan pinjaman.

#### 8. Perancangan Antarmuka *Form* Pengajuan Pinjaman

**Gambar 5.21 Perancangan Antarmuka Form Pengajuan Pinjaman**

Gambar 5.21 merupakan rancang antarmuka halaman *form* untuk pengajuan pinjaman. Anggota harus mengisi *form* yang ada untuk mengajukan pinjaman terdapat 3 tipe *form* yaitu *textfield*, *date* dan *textarea*. Selain itu terdapat tombol kirim untuk mengajukan permohonan pinjaman.

#### 9. Perancangan Antarmuka Konfirmasi Pengajuan

**Gambar 5.22 Antarmuka Konfirmasi Pengajuan**

Gambar 5.22 merupakan rancangan antarmuka petugas dimana dapat menampilkan informasi tentang daftar pengajuan pinjaman. Daftar ditampilkan dalam bentuk table dimana terdapat pilihan aksi yang dapat dilakukan oleh petugas.



## BAB 6 IMPLEMENTASI

Bab implementasi merupakan dokumentasi dari fase Konstruksi dalam metode metode *Relational Unified Proses (RUP)* yang didalamnya terdiri dari penjelasan tentang spesifikasi lingkungan pengembangan sistem, implementasi *class*, implementasi basis data dan implementasi antarmuka pengguna.

### 6.1 Spesifikasi Sistem

Spesifikasi sistem merupakan gambaran dari kebutuhan dalam pembuatan sistem. Spesifikasi tersebut dapat berupa spesifikasi perangkat lunak dan perangkat keras. Dimana dalam pengembangan sistem informasi *customer touching application* unit simpan pinjam KPRI “GURU” Kecamatan Sumbermanjing ini digunakan oleh peneliti dalam melakukan implementasi pada fase konstruksi metode RUP.

#### 6.1.1 Spesifikasi Perangkat Keras

Spesifikasi dari perangkat keras yang digunakan oleh peneliti dalam pengembangan sistem informasi *customer touching application* unit simpan pinjam KPRI “GURU” Kecamatan Sumbermanjing dapat dilihat pada table 6.1.

**Table 6.1 Spesifikasi Perangkat Keras**

Nama Komponen	Spesifikasi
<i>System Model</i>	Asus A46CB
<i>Processor</i>	Intel(R) Core I5-3317U @1.70GHz
<i>Memory</i>	500GB HDD
<i>Memory RAM</i>	8 GB

#### 6.1.2 Spesifikasi Perangkat Lunak

Spesifikasi dari perangkat lunak yang digunakan oleh peneliti dalam pengembangan sistem informasi *customer touching application* Unit Simpan Pinjam KPRI “GURU” Kecamatan Sumbermanjing dapat dilihat pada table 6.1.

**Table 6.2 Spesifikasi Perangkat Lunak**

Nama Komponen	Spesifikasi
<i>Operating System</i>	Windows 8 62-bit
<i>Program Language</i>	PHP, HTML, CSS, Javascript
<i>Framework</i>	Codeigniter
<i>Text Editor</i>	Sublime Text

Web Server	Apache
Database	MySQL
Browser	Google Chrome

## 6.2 Implementasi Class

Pada bab perancangan terdapat perancangan *class* yang diterapkan pada sistem yang dikembangkan. Penulis menerapkan *Object Oriented Programing* sehingga *class* terbagi menjadi *model*, *view*, dan *controller*. Berikut merupakan *class* yang diimplementasikan dalam sistem.

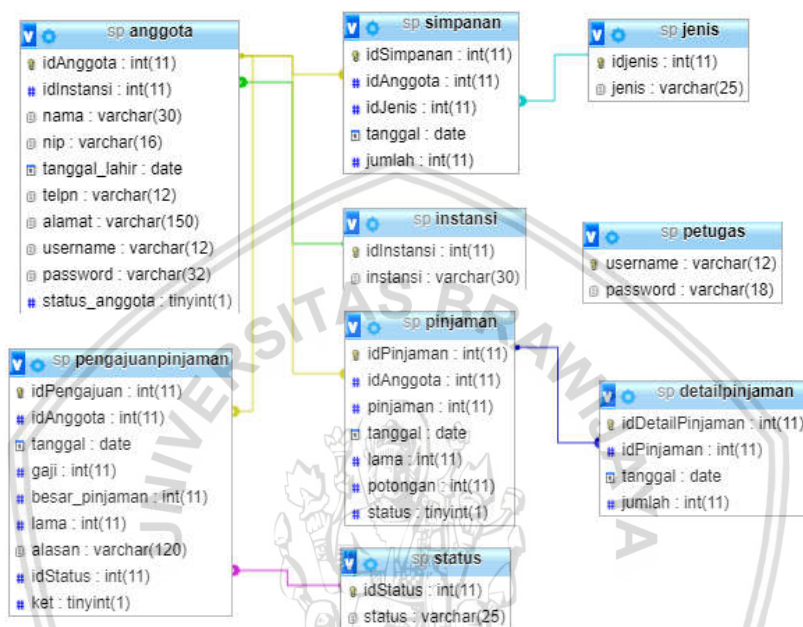
**Table 6.3 Implementasi Class**

No	Package	Nama Kelas	Nama File
1	<i>Models</i>	Mdl_anggota	Mdl_anggota
2	<i>Models</i>	Mdl_petugas	Mdl_petugas
3	<i>Models</i>	Mdl_simpanan	Mdl_simpanan
4	<i>Models</i>	Mdl_pinjaman	Mdl_pinjaman
5	<i>Models</i>	Mdl_pengajuan	Mdl_pengajuan
6	<i>Models</i>	Mdl_detailpinjaman	Mdl_detailpinjaman
7	<i>Models</i>	Mdl_instansi	Mdl_instansi
8	<i>Controller</i>	Ctrl_login	Ctrl_login
9	<i>Controller</i>	Crtl_anggotaP	Crtl_anggotaP
10	<i>Controller</i>	Crtl_anggotaA	Crtl_anggotaA
11	<i>Controller</i>	Crtl_pengajuanP	Crtl_pengajuanP
12	<i>Controller</i>	Crtl_pinjamanP	Crtl_pinjamanP
13	<i>Controller</i>	Crtl_simpananP	Crtl_simpananP
14	<i>Controller</i>	Crtl_pengajuanA	Crtl_pengajuanA
15	<i>Controller</i>	Crtl_pinjamanA	Crtl_pinjamanA
16	<i>Controller</i>	Crtl_simpananA	Crtl_simpananA

Table 6.3 merupakan daftar *class model* dan *controller* yang diimplementasikan dalam sistem informasi *customer-touching applications* Unit Simpan Pinjam KPRI “GURU” Kec. Sumbermanjing. Selain itu juga implementasi *class view* untuk menunjang tampilan dari informasi yang ada.

### 6.3 Implementasi Basis Data

Implementasi basis data dilakukan berdasarkan perancangan sistem yang diterapkan ke dalam basis data *MySQL*. Terdapat 9 table yang saling berhubungan dan telah dilakukan normalisasi *database*. Skema baris data yang telah diimplementasikan dapat dilihat pada Gambar 6.1.



Gambar 6.1 Implementasi Basis Data

### 6.4 Implementasi Algoritma

Implementasi algoritma berisikan beberapa kode program dari fungsi yang telah dilakukan rancangan algoritmanya pada tahap perancangan sistem. Pada implementasi algoritma terdapat beberapa fungsi utama dalam pengembangan sistem ini. Berikut merupakan beberapa implementasi fungsi sistem informasi *customer touching applications* Unit Simpan Pinjam KPRI.

#### 6.4.1 Algoritma Melihat Informasi Simpanan

Fungsi simpanan() terdapat pada *controller* *Ctrl\_simpananA* yang berfungsi untuk melihat informasi simpanan dari anggota. Table 6.4 merupakan baris kode program dari fungsi simpanan() tersebut.

Table 6.4 Kode Program Fungsi simpanan()

1	function simpanan(){
2	\$id = \$this->session->userdata("idUser");
3	\$data0 = \$this->Mdl_simpanan->getJumlah(\$id,0);
4	\$tarik = \$data0[0]['total'];
5	\$data1 = \$this->Mdl_simpanan->getJumlah(\$id,1);
6	\$pokok = \$data1[0]['total'];

```

7      $data2 = $this->Mdl_simpanan->getJumlah($id,2);
8      $wajib = $data2[0]['total'];
9      $data3 = $this->Mdl_simpanan->getJumlah($id,3);
10     $gertab = $data3[0]['total'];
11     $data4 = $this->Mdl_simpanan->getJumlah($id,4);
12     $thr = $data4[0]['total'];
13     $data5 = $this->Mdl_simpanan->getJumlah($id,5);
14     $manasuka = $data5[0]['total'];
15     $data6 = $this->Mdl_simpanan->getJumlah($id,6);
16     $shu = $data6[0]['total'];
17     $total = ($pokok+$wajib+$gertab+$thr+$manasuka+$shu)-
18     $tarik;
19     $data = array(
20         'pokok' => $pokok,
21         'wajib' => $wajib,
22         'gertab' => $gertab,
23         'thr' => $thr,
24         'manasuka' => $manasuka,
25         'total' => $total
26     );
27     $this->load->view('v_simpananA',$data);
28 }

```

#### 6.4.2 Algoritma Melihat Detail Simpanan

Fungsi detail\_simpanan() terdapat pada *controller* Ctrl\_simpananA yang berfungsi untuk melihat detail informasi simpanan dari anggota. Table 6.5 merupakan baris kode program dari fungsi detail\_simpanan() tersebut.

**Table 6.5 Kode Program Fungsi detail\_simpanan()**

```

1  function detail_simpanan(){
2      $id = $this->session->userdata("idUser");
3      $res = $this->Mdl_simpanan->getDetail($id);
4      $data = array();
5      $count = 0;
6      $total = 0;
7      foreach ($res as $k) {
8          if ($k['idJenis'] > 0) {
9              $total = $total + $k['jumlah'];
10             $data[$count] = array(
11                 'tanggal' => $k['tanggal'],
12                 'idJenis' => $k['idJenis'],
13                 'jenis' => $k['jenis'],
14                 'jumlah' => $k['jumlah'],
15                 'total' => $total
16             );
17             $count++;
18         }
19         else{
20             $total = $total - $k['jumlah'];
21             $data[$count] = array(
22                 'tanggal' => $k['tanggal'],
23                 'idJenis' => $k['idJenis'],
24                 'jenis' => $k['jenis'],
25                 'jumlah' => $k['jumlah'],
26                 'total' => $total
27             );
28             $count++;
29         }
30     }
31     $data1 = array_reverse($data);
32 }

```

33	<pre> \$this-&gt;load-&gt;view('v_detailSimpananA',array('data' =&gt; \$data1)); </pre>
----	---

#### 6.4.3 Algoritma Mengajukan pinjaman

Fungsi insertPengajuan () terdapat pada *controller* Ctrl\_pengajuanA yang berfungsi untuk mengajukan permohonan pinjaman. Table 6.6 merupakan baris kode program dari fungsi insertPengajuan() tersebut.

**Table 6.6 Kode Program Fungsi insertPengajuan()**

1	function insertPengajuan(){
2	\$id = \$this->session->userdata("idUser");
3	\$dataInsert = array(
4	'idPengajuan' => '',
5	'idAnggota' => \$id,
6	'gaji' => \$_POST['gaji'],
7	'besar_pinjaman' => \$_POST['pinjam'],
8	'lama'=> \$_POST['lama'],
9	'alasan'=> \$_POST['alasan'],
10	'idStatus'=> 1,
11	'ket' =>1
12	);
13	\$this->Mdl_pengajuan->insertData(\$dataInsert);
14	\$this->session->set_flashdata('sukses','Form       Pengajuan
15	Telah di Kirim');
16	redirect(base_url("index.php/Ctrl_pengajuanA/pengajuan"));
17	}

#### 6.4.4 Algoritma Melihat Detail Pengajuan

Fungsi detailPengajuan() terdapat pada *controller* Ctrl\_pengajuanA yang berfungsi untuk melihat detail dari pengajuan yang telah dilakukan. Table 6.5 merupakan baris kode program dari fungsi detailPengajuan() tersebut.

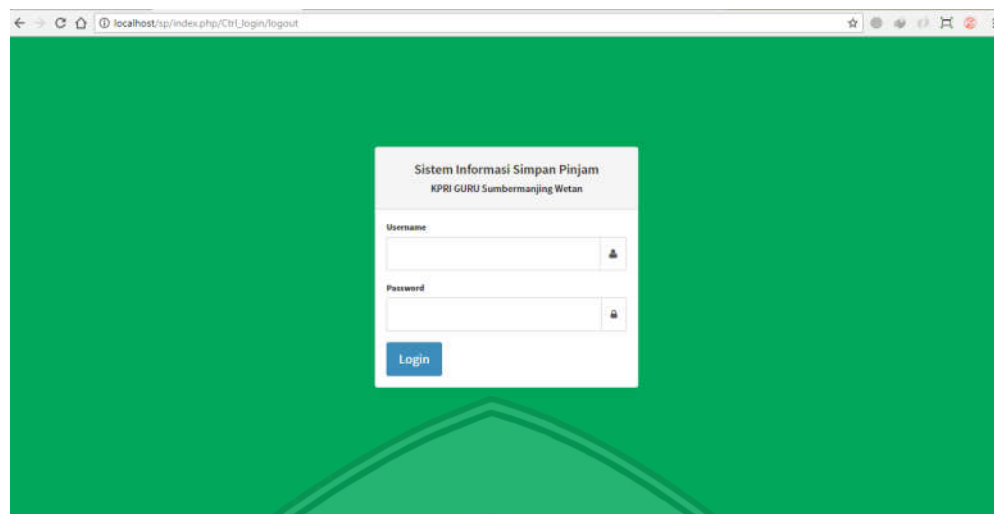
**Table 6.7 Kode Program Fungsi detailPengajuan()**

1	function detailPengajuan(\$id){
2	\$data2 = \$this->Mdl_pengajuan->getDetailpengajuan(\$id);
3	\$this->load->
4	view('v_detailPengajuanP',array('data'=>\$data2));

#### 6.5 Implementasi Antarmuka

Implementasi antarmuka pengguna dibuat berdasarkan perancangan yang diterapkan ke dalam bahasa *PHP* sebagai *view* pada *framework Codeigniter*. Berikut adalah hasil implementasi antarmuka pengguna.

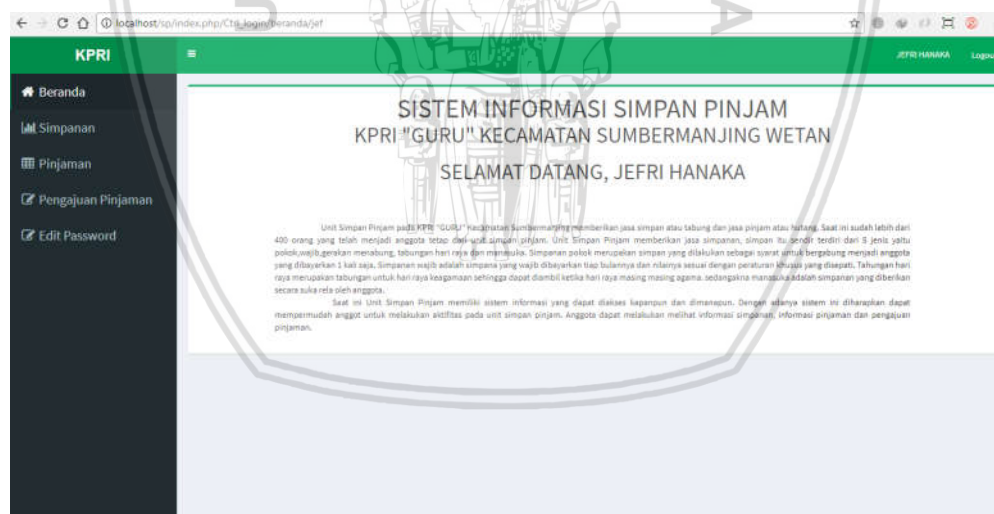
### 6.5.1 Implementasi Antarmuka *Login* Anggota



**Gambar 6.2 Antarmuka login anggota**

Pada gambar 6.2 menunjukkan hasil implementasi antarmuka *login* bagi anggota. Halaman *login* terdiri dari *form* berisikan kolom *username* dan *password*. Halaman ini merupakan halaman *default* dari sistem.

### 6.5.2 Implementasi Antarmuka *Home* Anggota



**Gambar 6.3 Antarmuka *Home* Anggota**

Pada gambar 6.3 menunjukkan hasil implementasi antarmuka *home* bagi anggota. Halaman ini menampilkan informasi dan ucapan kepada anggota. Terdapat informasi mengenai unit simpan pinjam dan sistem secara singkat.



### 6.5.3 Implementasi Antarmuka Simpanan Anggota

Tabung	M400	Gertak	T100	Manasuka	Total
8530000	100000	1547000	200000	0	10437000

**Gambar 6.4 Antarmuka simpanan anggota**

Pada gambar 6.4 menunjukkan hasil implementasi antarmuka simpanan bagi anggota. Halaman menampilkan tabel data dari simpanan yang dimiliki anggota. Data yang ditampilkan berupa jumlah dari setiap jenis simpanan beserta total simpanan yang dimiliki. Terdapat tombol detail simpanan untuk mengakses halaman detail simpanan.

### 6.5.4 Implementasi Antarmuka Detail Simpanan

No	Tanggal	Tabung	M400	Gertak	T100	Manasuka
1	20-12-2017	100000	0	0	0	0
2	20-12-2017	200000	0	200000	200000	0

No	Tanggal	Jumlah
1	20-12-2017	200000

**Gambar 6.5 Antarmuka Detail Simpanan**

Pada gambar 6.5 menunjukkan hasil implementasi antarmuka detail simpanan bagi anggota. Halaman ini terdiri dari 2 table yaitu menampilkan informasi detail penambahan dari nilai simpanan dan detail penarikan simpanan. Pada fase konstruksi terdapat perubahan tampilan yaitu dengan menggabungkan table penambahan dan penarikan menjadi 1 table seperti pada gambar 6.6.

**KPRI** JEFRI HANAKA Logout

Beranda Simpanan Pinjaman Pengajuan Pinjaman Edit Password

Riwayat Transaksi Simpanan

Data pada : 2018-01-05 01:43:58pm

No	Tanggal	Keterangan	Debit	Kredit	Jumlah
1	01-12-2017	Pokok	Rp. 100.000,-		Rp. 100.000,-
2	01-01-2017	Wajib	Rp. 150.000,-		Rp. 150.000,-
3	01-01-2017	Gertab	Rp. 30.000,-		Rp. 30.000,-
4	01-02-2017	Wajib	Rp. 150.000,-		Rp. 150.000,-
5	01-02-2017	Gertab	Rp. 30.000,-		Rp. 30.000,-
6	03-01-2017	Penarikan		Rp. 100.000,-	Rp. 100.000,-
7	02-01-2018	Wajib	Rp. 150.000,-		Rp. 150.000,-
8	02-01-2018	Manasuka	Rp. 4.000.000,-		Rp. 4.000.000,-
9	02-01-2018	Gertab	Rp. 10.000,-		Rp. 10.000,-
10	02-01-2018	Penarikan		Rp. 25.000,-	Rp. 25.000,-

Previous 1 Next

Gambar 6.6 Antarmuka Detail Simpanan (Fase Konstruksi)

### 6.5.5 Implementasi Antarmuka Pinjaman

**KPRI** JEFRI HANAKA Logout

Beranda Simpanan Pinjaman Pengajuan Pinjaman Edit Password

Pinjaman

Data pada : 2018-05-27 09:24:29pm

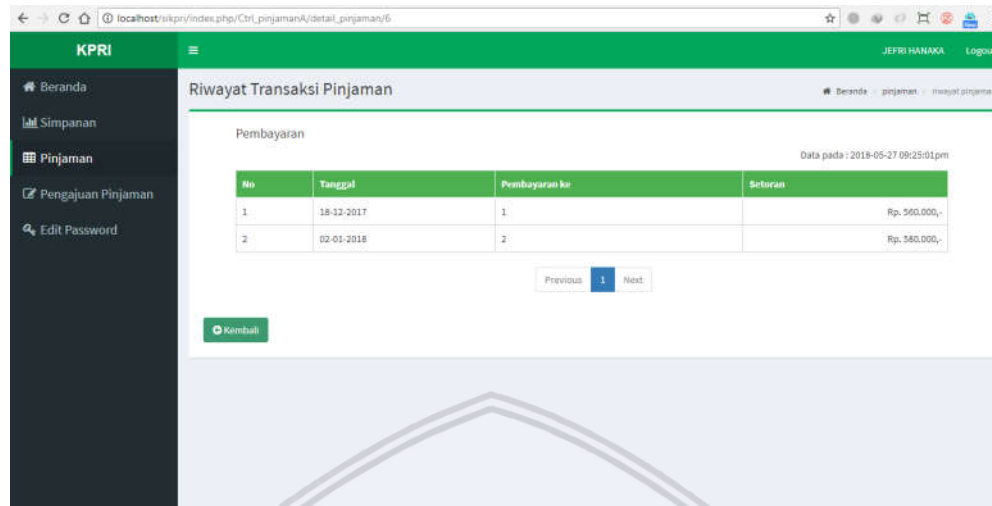
Tanggal Pinjam	Besar Pinjaman	Lama	Potongan	Angsuran Ke	Sisa Pokok
2017-12-12	Rp. 6.000.000,-	12	Rp. 560.000,-	2	Rp. 5.000.000,-

Detail Pinjaman

Gambar 6.7 Antarmuka pinjaman anggota

Pada gambar 6.7 menunjukkan hasil implementasi antarmuka pinjaman bagi anggota. Halaman menampilkan tabel data dari pinjaman yang dimiliki anggota. Data yang ditampilkan berupa jumlah sisa pinjaman dan jumlah setoran yang telah dilakukan. Terdapat tombol detail pinjaman untuk mengakses halaman detail pinjaman.

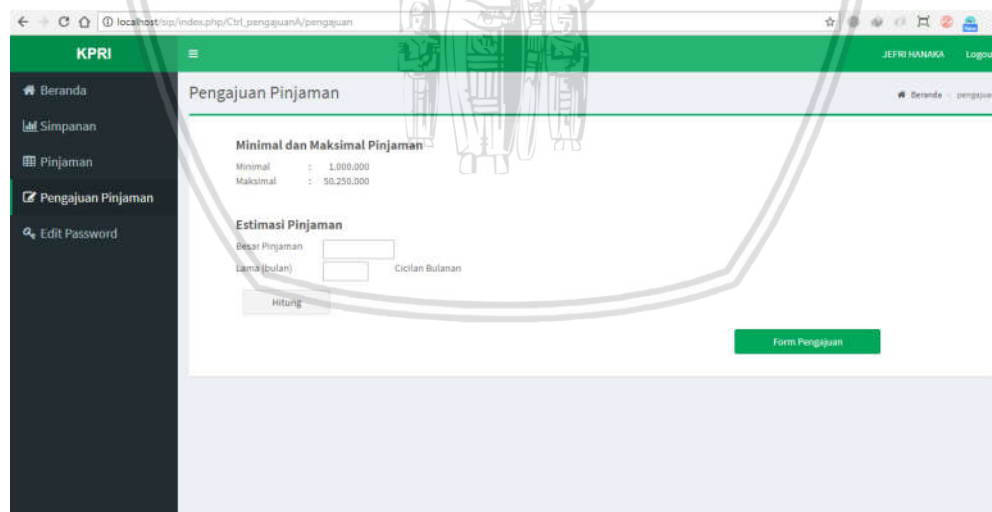
### 6.5.6 Implementasi Antarmuka Detail Pinjaman



**Gambar 6.8 Antarmuka detail pinjaman**

Pada gambar 6.8 menunjukkan hasil implementasi antarmuka detail pinjaman bagi anggota. Halaman menampilkan tabel data dari detail setoran pinjaman yang dilakukan oleh anggota. Informasi yang ditampilkan meliputi tanggal, setoran ke, dan jumlah setoran.

### 6.5.7 Implementasi Antarmuka Pengajuan Pinjaman



**Gambar 6.9 Antarmuka Pengajuan Pinjaman Fase Elaborasi**

Pada gambar 6.9 menunjukkan hasil implementasi antarmuka pengajuan pinjaman bagi anggota. Halaman menampilkan nilai minimum dan maksimum dari pinjaman yang dilayani oleh unit simpan pinjam. Terdapat form yang dapat melakukan perhitungan estimasi potongan/setoran tiap bulan berdasarkan jumlah pinjaman dan lamanya. Pada bagian pengajuan terdapat informasi

mengenai peraturan pinjaman yang ditetapkan unit simpan pinjam. Tombol isi form online berfungsi untuk menuju halaman form pengajuan.

Gambar 6.9 merupakan hasil dari antarmuka pengajuan pinjaman, terdapat perubahan pada fase konstruksi. Perubahan yang dilakukan adalah dengan menambahkan status dari pengajuan dan informasi tentang pengajuan pinjaman yang dilakukan. Setelah dilakukan perbaikan maka antarmuka pengajuan pinjaman seperti pada gambar 6.10.

Gambar 6.10 Antarmuka Pengajuan Pinjaman Fase Konstruksi

### 6.5.8 Implementasi Antarmuka Form Pengajuan

Gambar 6.11 Antarmuka Form Pengajuan

Gambar 6.11 merupakan hasil implementasi dari antarmuka form pengajuan pinjaman. Pada halaman ini terdapat form untuk mengajukan pinjaman, pada form tersebut juga terdapat beberapa field yang secara otomatis terisi dengan data dari anggota.

### 6.5.9 Implementasi Antarmuka Edit *Password*

**Gambar 6.12 Antarmuka Edit Password**

Gambar 6.12 merupakan antarmuka dari halaman edit *password*. Pada halaman ini terdapat form untuk melakukan perubahan *password* dari anggota.

### 6.5.10 Implementasi Antarmuka Detail Pengajuan

Pengajuan Pinjaman	
Nama	Jefri Hanaka
NIP	1985033020031220
Tanggal Lahir	1985-03-30
Instansi	SDN Sumbermarang 3
Alamat	Jln. apel batu
Nomor Telp	08526754687
Gaji	3000000
Pinjaman	10000000
Lama	12
Asas	biaya sekolah

**Gambar 6.13 Antarmuka Detail Pengajuan**

Gambar 6.13 merupakan antarmuka dari halaman detail pengajuan. Pada antarmuka tersebut terdapat informasi mengenai pengajuan yang sedang dilakukan oleh anggota. Informasi tersebut dapat membantu anggota untuk mengisi form cetak ketika pengajuan diterima.

## BAB 7 PENGUJIAN

Pengujian merupakan bagian dari tahap *Relation Unified Proses* yaitu termasuk dalam fase konstruksi dan fase transisi. Tujuan dari bab pengujian adalah untuk mengetahui bagaimana implementasi sistem apakah terdapat cacat atau *error* yang terjadi. Pengujian juga bertujuan untuk mengetahui apakah sistem yang telah dikembangkan sudah sesuai dan mampu menyelesaikan permasalahan yang ada. Selain itu pengujian bertujuan untuk mengetahui penerimaan sistem oleh pengguna akhir.

### 7.1 Pengujian Black-Box

Metode pengujian *black-box* fokus pengujian pada fungsional dari sistem. Selain itu pengujian dilakukan untuk mengetahui seberapa banyak kesalahan fungsi secara garis besar, antarmuka, akses database, inisialisasi dan tujuan akhir. Berikut hasil dari pengujian *black-box*.

#### 7.1.1 Perancangan Pengujian Black-Box

Rancangan pengujian *black-box* adalah pendefinisian dari kondisi dan hasil yang diharapkan dari sistem. Rancangan berfungsi sebagai acuan dalam proses pengujian dan sebagai acuan penilaian apakah hasil dari pengujian tersebut berhasil atau gagal.

Table 7.1 Perancangan Pengujian Black-Box

No	Kode Fungsi	Pengujian	Test case	Harapan
1	SIKPRI-KF-01	Login berhasil	Username dan password benar	Sistem menampilkan halaman <i>home</i>
		Username tidak terdaftar	Menginputkan <i>username</i> yang tidak terdaftar	Sistem menampilkan pesan <i>username</i> tidak terdaftar
		Username dan password kosong	Kolom <i>username</i> dan <i>password</i> tidak diisi	Sistem menampilkan pesan kolom harus diisi
		Username dan password salah	Menginputkan <i>username</i> atau <i>password</i> yang tidak sesuai	Sistem menampilkan peringatan <i>username</i> dan <i>password</i> salah
2	SIKPRI-KF-02	Logout berhasil	Menekan tombol <i>password</i>	Sistem menghapus



No	Kode Fungsi	Pengujian	Test case	Harapan
				session dan menampilkan halaman <i>login</i>
3	SIKPRI-KF-03	Menampilkan data simpanan	Terdapat data simpanan	Sistem menampilkan data simpanan pada table
		Tidak terdapat data simpanan	Tidak terdapat data pada <i>database</i>	Sistem menampilkan informasi bahwa nilai 0
4	SIKPRI-KF-20	Memiliki detail simpanan	Terdapat data pada detail simpanan	Sistem menampilkan data detail simpanan sesuai dengan anggota
5	SIKPRI-KF-04	Menampilkan data pinjaman	Terdapat data pinjaman	Sistem menampilkan data simpanan pada table
		Tidak terdapat data pinjaman	Tidak terdapat data pada database	Sistem menampilkan informasi bahwa nilai 0
6	SIKPRI-KF-21	Memiliki detail pinjaman	Terdapat data pada detail pinjaman	Sistem menampilkan data sesuai dengan anggota
		Tidak memiliki detail pinjaman	Tidak terdapat data detail pinjaman	Sistem menampilkan table dengan keterangan tidak terdapat data
7	SIKPRI-KF-05	Menampilkan pesan tidak memiliki pengajuan pinjaman	Tidak terdapat data pada database pengajuan pinjaman	Sistem menampilkan pesan tidak memiliki pengajuan dan tidak menampilkan

No	Kode Fungsi	Pengujian	Test case	Harapan
				tombol lihat detail pengajuan
		Menampilkan pesan pengajuan sedang di proses	Nilai kolom status pada database penajuan pinjaman bernilai 1	Sistem menampilkan pesan pengajuan dalam proses
		Menampilkan pesan pengajuan diterima	Nilai kolom status pada database penajuan pinjaman bernilai 2	Sistem menampilkan pesan pengajuan dalam diterima
		Menampilkan pesan pengajuan ditolak	Nilai kolom status pada database penajuan pinjaman bernilai 3	Sistem menampilkan pesan pengajuan dalam ditolak
		Menampilkan form pengajuan	Menekan tombol isi form online	Sistem menampilkan form pengajuan pinjaman
		Form pengajuan kosong	Terdapat kolom yang kosong dalam form pengajuan	Sistem menampilkan peringatan bahwa kolom harus diisi
		Pengajuan berhasil	Seluruh data pengajuan sesuai dengan format	Sistem menampilkan halaman pengajuan dan pesan pengajuan berhasil
8	SIKPRI-KF-22	Perhitungan estimasi gagal	Menginputkan nilai yang tidak sesuai dengan format	Sistem menampilkan pesan input tidak valid
		Perhitungan berhasil	Menginputkan nilai sesuai dengan format yang ada	Sistem menampilkan hasil perhitungan
9	SIKPRI-KF-23	Unduh form pengajuan	Menekan tombol unduh	Browser mengunduh file

No	Kode Fungsi	Pengujian	Test case	Harapan
10	SIKPRI-KF-25	Menampilkan detail pengajuan	Menekan tombol detail pengajuan	Sistem menampilkan data detail pengajuan
11	SIKPRI-KF-24	Input <i>password</i> lama benar	Menginputkan <i>password</i> yang sama dengan <i>password</i> lama	Sistem memperbarui data dan menampilkan pesan berhasil
		Input <i>password</i> lama salah	Menginputkan <i>password</i> lama yang tidak sesuai	Sistem menampilkan pesan <i>password</i> lama salah
		Form pinjaman baru kosong	Terdapat form yang kosong	Sistem menampilkan pesan form tidak boleh kosong
12	SIKPRI-KF-06	Melakukan konfirmasi diterima	Menekan tombol diterima	Sistem akan merubah tampilan status menjadi disetujui
		Melakukan konfirmasi ditolak	Menekan tombol ditolak	Sistem akan merubah tampilan status menjadi ditolak
		Melakukan konfirmasi pengajuan selesai	Menekan tombol selesai	Sistem akan merubah menghapus dari daftar pengajuan
13	SIKPRI-KF-07	Melihat data pengajuan	Menekan tombol detail	Sistem akan menampilkan informasi dari pengajuan pinjaman
14	SIKPRI-KF-08	Terdapat data simpanan	Ada beberapa data simpanan	Sistem menampilkan data simpanan dalam bentuk table

No	Kode Fungsi	Pengujian	Test case	Harapan
		Tidak ada data simpanan	Data simpanan kosong	Sistem menampilkan table dengan keterangan tidak terdapat data.
15	SIKPRI-KF-10	Menampilkan pilihan setoran	Menekan tombol update	Sistem menampilkan pilihan jenis simpanan
		Menambah simpanan wajib	Memilih jenis simpanan wajib dan menekan simpan	Sistem akan menampilkan data telah terupdate
		Menambah simpanan gertab	Memilih jenis simpanan gertab dan menekan simpan	Sistem akan menampilkan data telah terupdate
		Menambah simpanan THR	Memilih jenis simpanan thr dan menekan simpan	Sistem akan menampilkan data telah terupdate
		Menambah simpanan manasuka	Memilih jenis simpanan manasuka	Sistem menampilkan form jumlah simpanan manasuka
		Menambah manasuka berhasil	Menginputkan nilai dengan benar	Sistem menampilkan pesan data telah terupdate
		Menambah manasuka gagal	Menginputkan nilai 0 atau negatif	Sistem menampilkan peringatan bahwa nilai harus positif dan tidak boleh 0
		Menambah simpanan SHU	Memilih jenis simpanan SHU	Sistem menampilkan form jumlah simpanan SHU

No	Kode Fungsi	Pengujian	Test case	Harapan
		Menambah SHU gagal	Menginputkan nilai 0 atau negatif	Sistem menampilkan peringatan input tidak sesuai
		Melakukan penarikan berhasil	Menginputkan nilai dengan benar	Sistem menampilkan pesan data telah terupdate
		Menambah penarikan gagal	Menginputkan nilai 0 atau negatif	Sistem menampilkan peringatan bahwa nilai harus diatas 100000
16	SIKPRI-KF-12	Terdapat data pinjaman	Ada beberapa data pinjaman	Sistem menampilkan data pinjaman dalam bentuk table
		Tidak ada data pinjaman	Data pinjaman kosong	Sistem menampilkan table dengan keterangan tidak terdapat data.
17	SIKPRI-KF-13	Form pinjaman baru kosong	Terdapat form yang kosong	Sistem menampilkan peringatan form tidak boleh kosong
		Uji input negatif	Inputan berupa nilai negatif	Sistem menampilkan peringatan minimal 0
		Uji double pinjaman	Menginputkan pinjaman dengan anggota yang sama	Sistem menampilkan peringatan bahwa anggota masih memiliki simpanan aktif

No	Kode Fungsi	Pengujian	Test case	Harapan
		Uji tambah pinjaman berhasil	Menginputkan pinjaman dengan format yang benar dan anggota yg tidak memiliki pinjaman	Sistem menampilkan pesan pinjaman data telah tersimpan
18	SIKPRI-KF-14	Update setoran rutin	Memilih pilihan setoran rutin	Sistem menampilkan form tanggal
		Form tanggal kosong	Tidak mengisi form tanggal	Sistem menampilkan peringatan form tidak boleh kosong
		Update setoran rutin berhasil	Mengisi data pada kolom tanggal	Sistem menampilkan pesan data berhasil diupdate
		Update setoran tutup atau telat	Memilih pilihan setoran rutin	Sistem menampilkan form besar setoran
		Update setoran tutup atau telat berhasil	Mengisi data tanggal dan besar setoran sesuai dengan format	Sistem menampilkan pesan data berhasil diupdate
		Form setoran kosong	Form besar setoran kosong	Sistem menampilkan peringatan form tidak boleh kosong
		Uji input negatif	Inputan berupa nilai negatif	Sistem menampilkan peringatan minimal 0
19	SIKPRI-KF-15	Menghapus pinjaman	Menekan tombol selsesai	Sistem menampilkan pesan data berhasil di update



No	Kode Fungsi	Pengujian	Test case	Harapan
20	SIKPRI-KF-16	Terdapat data anggota	Ada beberapa data anggota	Sistem menampilkan data anggota dalam bentuk table
		Tidak ada data anggota	Data anggota kosong	Sistem menampilkan table dengan keterangan tidak terdapat data.
21	SIKPRI-KF-17	Menambah anggota	Mengisi form secara lengkap dan benar	Sistem menampilkan pesan data berhasil di tambahkan
		Form anggota baru kosong	Form data anggota kosong	Sistem menampilkan peringatan form tidak boleh kosong
		Input NIP yang sama	Menginputkan nilai NIP yang sama	Sistem menampilkan pesan anggota telah terdaftar menjadi anggota non aktif
22	SIKPRI-KF-18	Merubah data anggota	Merubah input form anggota dan menekan tombol simpan	Sistem menampilkan pesan data anggota telah diupdate
		Form update anggota kosong	Form data anggota kosong	Sistem menampilkan peringatan form tidak boleh kosong
23	SIKPRI-KF-19	Menghapus data anggota	Menekan tombol non aktif	Sistem menampilkan pesan berhasil

No	Kode Fungsi	Pengujian	Test case	Harapan
				dan data hilang dari table

### 7.1.2 Hasil Pengujian *Black-box*

Analisa dari hasil penerapan perancangan pengujian *black-box*. Teknik analisa yaitu membandingkan harapan yang telah didefinisikan pada perancangan dengan hasil yang ditampilkan oleh sistem. Berikut merupakan hasil dari pengujian *black-box*.

**Table 7.2 Table hasil pengujian *black box***

No	Pengujian	Harapan	Hasil
1	<i>Username</i> dan <i>password</i> benar	Sistem menampilkan halaman <i>home</i>	Berhasil
2	Menginputkan <i>username</i> yang tidak terdaftar	Sistem menampilkan pesan <i>username</i> tidak terdaftar	Berhasil
3	Kolom <i>username</i> dan <i>password</i> tidak diisi	Sistem menampilkan pesan kolom harus diisi	Berhasil
4	Menginputkan <i>username</i> atau <i>password</i> yang tidak sesuai	Sistem menampilkan peringatan <i>username</i> dan <i>password</i> salah	Berhasil
5	Menekan tombol <i>password</i>	Sistem menghapus <i>session</i> dan menampilkan halaman <i>login</i>	Berhasil
6	Terdapat data simpanan	Sistem menampilkan data simpanan pada table	Berhasil
7	Tidak terdapat data pasda database	Sistem menampilkan informasi bahwa nilai 0	Berhasil
8	Terdapat data pada detail simpanan	Sistem menampilkan data detail simpanan sesuai dengan anggota	Berhasil

No	Pengujian	Harapan	Hasil
9	Terdapat data pinjaman	Sistem menampilkan data simpanan pada table	Berhasil
10	Tidak terdapat data pada database	Sistem menampilkan informasi bahwa nilai 0	Berhasil
11	Terdapat data pada detail pinjaman	Sistem menampilkan data sesuai dengan anggota	Berhasil
12	Tidak terdapat data detail pinjaman	Sistem menampilkan table dengan keterangan tidak terdapat data	Berhasil
13	Tidak terdapat data pada database pengajuan pinjaman	Sistem menampilkan pesan tidak memiliki pengajuan dan tidak menampilkan tombol lihat detail pengajuan	Berhasil
14	Nilai kolom status pada database penajuan pinjaman bernilai 1	Sistem menampilkan pesan pengajuan dalam proses	Berhasil
15	Nilai kolom status pada database penajuan pinjaman bernilai 2	Sistem menampilkan pesan pengajuan dalam diterima	Berhasil
16	Nilai kolom status pada database penajuan pinjaman bernilai 3	Sistem menampilkan pesan pengajuan dalam ditolak	Berhasil
17	Menekan tombol isi form <i>online</i>	Sistem menampilkan form pengajuan pinjaman	Berhasil
18	Terdapat kolom yang kosong dalam form pengajuan	Sistem menampilkan peringatan bahwa kolom harus diisi	Berhasil
19	Menginputkan nilai yang benar pada form pengajuan	Sistem menampilkan pengajuan berhasil	Berhasil
20	Menginputkan nilai sesuai format	Sistem menampilkan hasil perhitungan estimasi	Berhasil

No	Pengujian	Harapan	Hasil
21	Menginputkan nilai yang tidak sesuai format	Sistem menampilkan pesan input tidak valid	Berhasil
22	Menekan tombol unduh	<i>Browser</i> mengunduh file	Berhasil
23	Menampilkan detail pengajuan	Sistem menampilkan data detail pengajuan	Berhasil
24	Menginputkan <i>password</i> yang sama dengan <i>password</i> lama	Sistem memperbarui data dan menampilkan pesan berhasil	Berhasil
25	Menginputkan <i>password</i> lama yang tidak sesuai	Sistem menampilkan pesan <i>password</i> lama salah	Berhasil
26	Terdapat form yang kosong	Sistem menampilkan pesan form tidak boleh kosong	Berhasil
27	Menginputkan <i>password</i> lama yang tidak sesuai	Sistem menampilkan pesan <i>password</i> lama salah	Berhasil
28	Terdapat form yang kosong	Sistem menampilkan pesan form tidak boleh kosong	Berhasil
29	Menekan tombol diterima	Sistem akan merubah tampilan status menjadi disetujui	Berhasil
30	Menekan tombol ditolak	Sistem akan merubah tampilan status menjadi ditolak	Berhasil
31	Menekan tombol selesai	Sistem akan merubah menghapus dari daftar pengajuan	Berhasil
32	Melihat data pengajuan	Sistem akan menampilkan informasi dari pengajuan pinjaman	Berhasil
33	Ada beberapa data simpanan	Sistem menampilkan data simpanan dalam bentuk table	Berhasil

No	Pengujian	Harapan	Hasil
34	Data simpanan kosong	Sistem menampilkan table dengan keterangan tidak terdapat data.	Berhasil
35	Menekan tombol update	Sistem menampilkan pilihan jenis simpanan	Berhasil
36	Memilih jenis simpanan wajib dan menekan simpan	Sistem akan menampilkan data telah terupdate	Berhasil
37	Memilih jenis simpanan gertab dan menekan simpan	Sistem akan menampilkan data telah terupdate	Berhasil
38	Memilih jenis simpanan thr dan menekan simpan	Sistem akan menampilkan data telah terupdate	Berhasil
39	Memilih jenis simpanan manasuka	Sistem menampilkan form jumlah simpanan manasuka	Berhasil
40	Menginputkan nilai dengan benar	Sistem menampilkan pesan data telah terupdate	Berhasil
41	Menginputkan nilai 0 atau negatif	Sistem menampilkan peringatan bahwa nilai harus positif dan tidak boleh 0	Berhasil
42	Memilih jenis simpanan SHU	Sistem menampilkan form jumlah simpanan SHU	Berhasil
43	Menginputkan nilai 0 atau negatif	Sistem menampilkan peringatan bahwa nilai harus positif dan tidak boleh 0	Berhasil
44	Menginputkan nilai dengan benar	Sistem menampilkan pesan data telah terupdate	Berhasil
45	Menginputkan nilai 0 atau negatif	Sistem menampilkan peringatan bahwa nilai harus positif dan tidak boleh 0	Berhasil

No	Pengujian	Harapan	Hasil
46	Ada beberapa data pinjaman	Sistem menampilkan data pinjaman dalam bentuk table	Berhasil
47	Data pinjaman kosong	Sistem menampilkan table dengan keterangan tidak terdapat data.	Berhasil
48	Terdapat form yang kosong	Sistem menampilkan peringatan form tidak boleh kosong	Berhasil
49	Inputan berupa nilai negatif	Sistem menampilkan peringatan minimal 0	Berhasil
50	Menginputkan pinjaman dengan anggota yang sama	Sistem menampilkan pesan pinjaman data telah tersimpan	Berhasil
51	Menginputkan pinjaman dengan format yang benar dan anggota yg tidak memiliki pinjaman	Sistem menampilkan data berhasil disimpan	
52	Memilih pilihan setoran rutin	Sistem menampilkan form tanggal	Berhasil
53	Tidak mengisi form tanggal	Sistem menampilkan peringatan form tidak boleh kosong	Berhasil
54	Mengisi data pada kolom tanggal	Sistem menampilkan pesan data berhasil diupdate	Berhasil
55	Memilih pilihan setoran rutin	Sistem menampilkan form besar setoran	Berhasil
56	Mengisi data tanggal dan besar setoran sesuai dengan format	Sistem menampilkan pesan data berhasil diupdate	Berhasil
57	Form besar setoran kosong	Sistem menampilkan peringatan form tidak boleh kosong	Berhasil
58	Inputan berupa nilai negatif	Sistem menampilkan peringatan minimal 0	Berhasil



No	Pengujian	Harapan	Hasil
59	Menekan tombol selsesai	Sistem menampilkan pesan data berhasil di update	Berhasil
60	Ada beberapa data anggota	Sistem menampilkan data anggota dalam bentuk table	Berhasil
61	Data anggota kosong	Sistem menampilkan table dengan keterangan tidak terdapat data.	Berhasil
62	Mengisi form secera lengkap dan benar	Sistem menampilkan pesan data berhasil di tambahkan	Berhasil
63	Form data anggota kosong	Sistem menampilkan peringatan form tidak boleh kosong	Berhasil
64	Menginputkan nilai NIP yang sama	Sistem menampilkan pesan anggota telah terdaftar menjadi anggota non aktif	Berhasil
65	Merubah input form anggota dan menekan tombol simpan	Sistem menampilkan pesan data anggota telah diupdate	Berhasil
66	Form data anggota kosong	Sistem menampilkan peringatan form tidak boleh kosong	Berhasil
67	Menekan tombol non aktif	Sistem menampilkan pesan berhasil dan data hilang dari table	Berhasil

## 7.2 Pengujian *Compatibility*

Pengujian *compatibility* menggunakan perangkat lunak *sortsite*. Hasil dari pengujian menunjukkan bahwa terdapat *critical issues* pada *browser firefox* dan *chrome* hal ini dikarenakan tidak diaktifkannya fungsi *ssl* pada *apache* yang digunakan. Selain itu *minor issues* terdapat pada *Internet Explorer* versi 9 dimana itu dikarenakan versi browser tersebut tidak dapat mensupport beberapa fungsi *CSS*.

Browser Version	IE			Edge	Firefox	Safari		Opera	Chrome	iOS			Android		BlackBerry	
	9	10	11		58	≤ 10	11		63	≤ 9	10	11	≤ 3	4*	≤ 7.1	10.0
Critical Issues	✓	✓	✓	✓	●	✓	✓	✓	●	✓	✓	✓	✓	✓	✓	✓
Major Issues	✓	✓	✓	✓		✓	✓	✓		✓	✓	✓	✓	✓	✓	✓
Minor Issues	●	✓	✓	✓		✓	✓	✓		✓	✓	✓	✓	✓	✓	✓

Gambar 7.1 Hasil Pengujian *Compatibility*

### 7.3 Pengujian UAT

*User Acceptance Testing* (UAT) bertujuan untuk mengetahui bagaimana response pengguna kepada sistem informasi unit simpan pinjam KPRI. Pada pengujian ini responden tidak memperhatikan baris kode pada sistem, melainkan hanya memperhatikan apakah sistem tersebut sesuai dengan apa yang diinginkan oleh pengguna. Pengujian UAT bertujuan untuk mendapatkan kepercayaan dan gambaran bagaimana sistem akan diterima oleh pengguna akhir yaitu anggota unit simpan pinjam.

Dari beberapa kriteria yang dimiliki oleh pengujian UAT penulis memilih 4 kriteria dalam pengujian sistem informasi unit simpan pinjam KPRI Kec. Sumbermanjing Wetan. Empat kriteria tersebut antara lain *Performance*, *Usability*, *Functional Correctness and Completeness*, dan *Accuracy* karena dinilai mampu mewakili apakah sistem tersebut dapat diterima oleh pengguna atau tidak. Pengujian ini dilakukan kepada pengguna akhir atau anggota karena pada pembangunan sistem informasi ini berfokus kepada *customer* yang dalam hal ini adalah anggota.

Pelaksanaan pengujian *user acceptance testing* dilakukan dengan memberikan penjelasan mengenai sistem yang akan diuji untuk memberikan gambaran awal bagaimana sistem tersebut. Langkah berikutnya adalah meminta responden melakukan serangkaian *task* yang telah diberikan. Responden tidak diperkenankan untuk melakukan *task* diluar dari skenario yang diberikan. Responden memberikan penilaian terhadap pernyataan yang telah disiapkan setelah melakukan rangkaian *task* yang diberikan. Berikut merupakan skenario *task* pengujian UAT.

Table 7.3 Skenario *task* pengujian UAT

No	Tugas yang dikerjakan
1	Melakukan login untuk masuk kedalam sistem
2	Memilih menu simpanan untuk melihat informasi simpanan
3	Menekan tombol detail simpanan untuk melihat detail dari simpanan anggota.
4	Memilih menu pinjaman

5	Menekan tombol detail pinjaman
6	Memilih menu pengajuan pinjaman
7	Membaca informasi minimal dan maksimal pinjaman yang dapat diajukan
8	Melakukan perhitungan estimasi dengan memasukkan nominal dan lama pinjaman
9	Menekan tombol isi form online
10	Melihat detail pengajuan
11	Mengisi form dengan lengkap
12	Memilih menu edit password
13	Melakukan logout atau keluar dari sistem

Setelah melakukan serangkaian *task* diatas responden diminta untuk mengisi 8 pertanyaan yang telah disiapkan sesuai dengan pengalaman mencoba beberapa *task* tersebut. Delapan pertanyaan yang diberikan berdasar pada 4 kriteria yang ada pada UAT. Semua pertanyaan tersebut dapat diberi tidak setuju dan setuju mengacu pada skala *guttman*. Berikut merupakan pertanyaan pengujian UAT.

**Table 7.4 Kriteria dan pertanyaan UAT**

No	Kriteria	Definisi	Pertanyaan
1	<i>Performance</i>	Menggambarkan performa dari sistem yang ada dengan mendefinisikan kedalam parameter yang dapat diukur.	Sistem Informasi Simpan Pinjam KPRI dapat diakses dengan cepat?
2	<i>Usability</i>	Untuk mengetahui apakah sistem mudah untuk digunakan dan dipelajari.	Anggota dapat memperoleh informasi simpan pinjam KPRI pada sistem informasi dengan mudah? Anggota merasa senang dengan tampilan Sistem Informasi simpan pinjam?
3	<i>Functional Correctness and Completeness</i>	Menggambarkan bagaimana sistem yang telah dibuat sudah sesuai dengan apa yang diharapkan oleh pengguna dan telah memenuhi kebutuhan yang didefinisikan.	Adanya Sistem Informasi Simpan Pinjam mempermudah aktifitas dalam unit simpan pinjam? Adanya fitur estimasi pinjaman

			mempermudah anggota untuk menentukan pengajuan pinjaman ?
			Adanya fitur pengisian form pinjaman online mempermudah anggota dalam melakukan pengajuan pinjaman ?
			Sistem Informasi simpan pinjam meningkatkan pelayanan kepada anggota ?
4	Accuracy	Akurasi digunakan untuk mengetahui sejauh mana tingkat ketepatan nilai yang dihitung dengan nilai yang diharapkan.	Fitur estimasi mempunyai tingkat akurasi mendekati perhitungan sebenarnya?

Setelah dilakukan proses pengujian UAT oleh anggota unit simpan pinjam sebanyak 20 orang. Langkah selanjutnya adalah melakukan analisis terhadap hasil pengujian *user accepted testing*. Berikut merupakan hasil jawaban dari pertanyaan dari proses pengujian oleh responden yaitu anggota unit simpan pinjam KPRI “GURU”.

**Table 7.5 Hasil Wawancara User Acceptance Testing**

No	Kriteria	Jumlah pertanyaan	Jumlah Jawaban	
			Tidak Setuju	Setuju
1	<i>Performance</i>	20	0	20
2	<i>Usability</i>	40	11	29
3	<i>Functional Correctness and Completeness</i>	80	13	63
4	Accuracy	20	3	17
Total		160	23	137

Pada table 7.5 kriteria *performance* mendapatkan jawaban setuju sebanyak 20 atau 100% dari total jawaban yang didapatkan pada pengujian UAT. Hasil ini dipengaruhi dengan penggunaan server lokal saat pengujian berlangsung. Pada

kriteria *Usability* yang mempunyai total 40 pertanyaan terdapat total 11 pertanyaan yang tidak disetujui oleh responden yaitu 3 kemudahan mendapat informasi dan 8 pada tampilan sistem. Tampilan sistem yang masih awam dan ketika sistem tersebut diakses dengan resolusi layar yang kecil beberapa responden merasa tidak nyaman. Sedangkan 29 jawaban setuju yang berarti 72,5% dari total responden telah menerima sistem dari segi *usability*.

Untuk kriteria ke tiga *functional correctness and completeness* dari 4 pertanyaan atau 80 jawaban responden terdapat 13 jawaban tidak setuju atau 16,25%. Jawaban responden yang tidak setuju meliputi 3 pada pernyataan bahwa sistem mempermudah aktifitas simpan pinjam dan 3 pada pernyataan fitur estimasi pinjaman mempermudah anggota dalam pengajuan pinjaman. Sedangkan 7 jawaban tidak setuju dari responden mengenai fitur pengisian form secara online. Sehingga untuk kriteria *functional correctness and completeness* terdapat 67 jawaban atau 83,75% responden menerima sistem yang telah dikembangkan.

Pada kriteria *accuracy* terdapat 3 jawaban tidak setuju atau 15% dari total jawaban. Hal ini dikarenakan responden menilai pembulatan dalam perhitungan estimasi sesuai pembulatan yang diterapkan dalam proses simpan pinjam. Sedangkan sisa responden menjawab bahwa perhitungan estimasi pinjaman yang ada pada sistem informasi simpan pinjam dapat diterima yaitu sebesar 85% dari total jawaban responden atau 17 dari total 20 jawaban responden.

Sehingga hasil dari pengujian *user accepted testing* yang melibatkan 20 responden dengan total 160 jawaban. Dapat diambil kesimpulan bahwa dari pengujian UAT ini sebesar 83,125% menyatakan bahwa Sistem Informasi *Customer - Touching Applications* Unit Simpan Pinjam KPRI "GURU" Kec. Sumbermanjing dapat diterima. Dari hasil pengujian ini juga dapat dijadikan acuan bahwa strategi CRM yang diterapkan dapat berjalan dan mendapatkan respon yang positif dari anggota dimana bertindak sebagai *customer* yang dipertahankan.

## BAB 8 KESIMPULAN DAN SARAN

### 8.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut :

1. Pada fase inepsi perancangan proses bisnis menghasilkan usulan proses bisnis yang mempermudah aktivitas dalam unit simpan pinjam. Hal ini dikarenakan dengan adanya sistem yang dikembangkan dapat mengautomatisasi beberapa proses sehingga dapat menyelesaikan permasalahan. Berdasar pada analisis permasalahan yang ada maka pada fase ini terdapat 19 kebutuhan sistem yang diprioritaskan pada kebutuhan anggota berdasar pada jenis aplikasi *customer touching applications*.
2. Proses perancangan yang merupakan bagian dari fase elaborasi menghasilkan *use case*, *class diagram*, *sequence diagram*, *physical data model*, perancangan algoritma dan perancangan antarmuka. Tahap perancangan menghasilkan sebuah rancangan sistem informasi *customer touching applications* yang terdiri dari 9 *class controller* dan 7 *class model*. Pada fase elaborasi ini terdapat sebuah iterasi yang menyebabkan terjadinya 5 penambahan dan 2 pengurangan kebutuhan sistem, sehingga pada fase elaborasi terdapat 22 kebutuhan fungsional.
3. Fase konstruksi yang didalamnya terdapat implementasi dan pengujian sistem. Proses implementasi menghasilkan sebuah sistem yang berfokus pada pengguna anggota. Dalam fase konstruksi terdapat literasi yang menghasilkan perubahan tampilan dan penambahan kebutuhan sistem sehingga pada akhir pengembangan terdapat 23 kebutuhan fungsional. Pada tahap ini terdapat pengujian *blackbox* untuk menguji kebutuhan fungsional sistem dilakukan dengan melakukan 67 *test case* dengan hasil yang baik. Selain itu juga terdapat pengujian *compatibility* yang menunjukkan terdapat 2 *critical issue* dikarenakan tidak diaktifkannya fungsi ssl.
4. Pada fase transisi terdapat *user accepted testing*. Pengujian UAT dilakukan kepada 20 orang yang merupakan anggota unit simpan pinjam. Pengujian yang dilakukan dengan memberikan 8 pertanyaan dari 4 kriteria uji UAT menunjukkan bahwa sebesar 83,123% koresponden menerima sistem informasi *customer touching applications* unit simpan pinjam KPRI "GURU" Kec. Sumbermanjing Wetan dan hasil ini juga mewakili bahwa penerapan strategi CRM dapat diterima dan mendapatkan respon baik oleh pelanggan.



## 8.2 Saran

Saran yang dapat diberikan setelah melakukan penelitian pengembangan sistem *customer touching applications* Unit Simpan Pinjam KPRI “GURU” Kec. Sumbermanjing Wetan ini adalah :

1. Pada pengembangan sistem lebih lanjut dapat mengintegrasikan sistem dengan sistem lain untuk meningkatkan manfaat bagi unit simpan pinjam KPRI Kec. Sumbermanjing Wetan misalnya mengintegrasikan sistem unit simpan pinjam dengan unit lainnya sehingga data lebih terintegrasi dan dapat diolah dengan maksimal.
2. Dalam implementasi lebih lanjut dapat meningkatkan keamanan sistem informasi sehingga dapat memberikan jaminan terhadap data dari setiap pengguna.



## DAFTAR PUSTAKA

- Agiledata, 2018. *Agile/Evolutionary Data Modeling: From Domain Modeling to Physical Modeling*. [online] Tersedia di: <  
<http://agiledata.org/essays/agileDataModeling.html> > [Diakses 12 November 2017].
- Arifin, D. 2005. "Rancangan Aplikasi e-CRM pada PT Essensilindo Perdana". *Soft Computing, Intelligent Systems and Information Technology*.
- Paryitno,E., Amylia,N. 2017. "Positive Impact of CRM Implementation to Improving The Services of Animal Polyclinics Customers" . IEEE.
- Becker, J., Kugeler, M. & Rosemann, M., 2013. *Process Management: A Guide for the Design of Business Processes*. Berlin: Springer-Verlag.
- Bittner, K. & Spence, I., 2002. *Use Case Modeling*. Boston: Addison Wesley.
- Booch, G., Rumbaugh, J. & Jacobson, I., 1998. *The Unified Modeling Language User Guide*. Boston: Addison Wesley.
- Booch, G., Rumbaugh, J. & Jacobson, I., 2007. *Object-Oriented Analysis and Design with Applications*. Boston: Addison Wesley.
- Buttle, Francis., 2009. *Customer Relationship Management: Concepts and Technology*, 2nd ed., Butterworth-Heinemann, Burlington.
- IBM, D. T., 1998. *Rational Unified Process, Best Practices for Software*. Cupertino: Rational Software.
- Jalote,P. 2008. *A Concise Introduction to Software Engineering*. Dehlhi: Department of Computer Science and Engineering IIT.
- Kalakota,R. & Robinson,M., 2001. *Roadmap for Success. e-Business 2.0*. Boston: Addison Wesley.
- Kinasih,A. 2017. "Pengembangan Sistem Evaluasi Pembelajaran PAUD (Studi Kasus di PAUD Seruni 05 Kota Malang)". J-PTIHK.
- Kshirasagar,N.,Tripathy,P. 2008. *Software Testing And Quality Assurance*. Boston: Addison Wesley.
- Leffingwell, D. & Widrig, D., 2003. *Managing Software Requirements: A Use Case Approach*. 2nd penyunt. Boston: Addison Wesley.
- Lucidchart, 2012. *Business Process Modeling Software*. [online] Tersedia di : <  
<https://www.lucidchart.com/pages/business-process-modeling/software>>  
 [Diakses 26 Oktober 2017].
- Object Management Group, 2011. *Business Process Model and Notation (BPMN)*. [pdf] Object Management Group.
- Paryitno,E., Amylia,N. 2017. "Positive Impact of CRM Implementation to Improving The Services of Animal Polyclinics Customers" . IEEE.

- Payne, A., 2005. *Handbook of CRM*. Oxford: Butterworth-Heinemann.
- Pressman, R. S., 2010. *Software Engineering: A Practitioner's Approach*. 7th ed. New York: McGraw-Hill.
- Przybyłek, A., 2011. *Bridging the gap between business process models and use-case models*. Gdańsk, Gdańsk University of Technology.
- Reiner,K., 2010. *Introduction to Information Systems: Enabling and Transforming Business*. [e-book] John Wiley & Sons. Tersedia di: Google Books <<https://books.google.co.id/books?id=maK8xskEduYC>>[Diakses 21 Oktober 2017]
- Salehi,S. 2015. *Evaluation of the Effects of e-CRM on Customer Loyalty (Case Study: Esfahan Branch's of Sepah Bank)*. IEEE
- Sitio,A.& Tamba,H.,2001. *Koperasi: Teori dan Praktek*. Jakarta: Erlangga.
- Silver, B., 2012. *BPMN 2.0 Handbook*. 2<sup>nd</sup> ed. Florida: FutStrat.
- Sukamto, R.A. & Shalahuddin, M., 2014. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. 2nd ed. Bandung: Informatika.
- Sommerville, I., 2011. *Software Engineering*. 9th ed. Boston: Addison-Wesley.
- Taufik,M. 2015. "Analisis dan Implementasi Perancangan Metode Rational Unified Process pada layanan SDB dan Metode Pengujian Product Metric pada Bank Mandiri Cabang Palu Sam Ratulangi". *e-Proceeding of Engineering telkomuniversity.ac.id*.
- Zulfata,G.2017 "Pengembangan Electronic Customer Relationship Management (E-CRM) Pada Toko Sepatu Aneka Sport Malang Dengan Metode Kerangka Kerja Dynamic CRM". J-PTIIK.

## LAMPIRAN WAWANCARA

### LAMPIRAN

#### Hasil Wawancara

Apa saja jasa yang ditawarkan oleh unit simpan pinjam KPRI Kecamatan Sumbermanjing Wetan?

**Jawab :**

Saat ini unit simpan pinjam melayani simpanan atau biasa disebut tabungan dan pinjaman atau kredit. Untuk simpanan, terdapat 4 jenis simpanan yang pertama yaitu pokok dibayar 1 kali ketika daftar sebesar Rp. 100.000 ribu. Wajib dan gerakan menabung (gertab) yang dibayarkan rutin perbulannya masing – masing Rp. 150.000 dan Rp. 30.000. Tabungan hari raya atau THR yang dibayarkan 1 tahun sekali sebesar Rp. 200.000 . Yang terakhir manasuka simpanan sukarela oleh anggota.

Bagaimana proses bisnis atau aktivitas untuk jenis simpanana saat ini?

**Jawab :**

Untuk proses pendaftaran anggota, calon anggota harus datang langsung ke kantor dan memenuhi persyaratan. Setelah anggota terdaftar aktivitas simpanan dapat dilakukan melalui perwakilan KPRI yaitu bendahara tiap instansi. Jadi untuk pembayaran simpanan wajib, gertab dan thr sudah dikordinasikan melalui potong gaji. Tapi untuk manasuka tidak bisa melalui perwakilan, anggota sendiri yang harus datang.

Bagaimana proses bisnis atau aktivitas jenis jasa pinjaman saat ini?

**Jawab :**

Sebelum mendapatkan pinjaman terdapat proses pengajuan pinjaman. Jadi alurnya itu anggota datang ke kantor menemui petugas untuk minta formulir dan berkonsultasi mengenai gambaran pinjaman yang akan diajukan. Setelah itu biasanya mereka pulang untuk menentukan besar pinjaman dan lama. Setelah mengisi formulir mereka minta tanda tangan dari bendahara atau ketua kelompok baru setelah itu mengumpulkan formulir ke kantor. Kita memiliki batas pengumpulan tiap bulannya tanggal 15 karena tanggal 15 sampai 21 dilakukan sidang pengurus untuk menentukan pengajuan yang ada. Diatas tanggal 21 biasanya pemohon akan datang untuk melihat hasilnya kalau diterima bisa langsung mengajukan proses pencairan dana. Untuk pembayaran setoran perbulan bisa dari potong gaji atau datang ke kantor untuk tutup pinjaman.

Apakah ada syarat tertentu untuk mendapatkan pinjaman?

**Jawab :**

Iya ada, sudah disepakati bersama dalam persus (peraturan khusus). Misalnya anggota minimal bergabung selama 3 bulan dan tidak memiliki pinjaman.

Siapa saja yang terlibat dalam aktivitas pada unit simpan pinjam?

**Jawab :**

*Pertama yang jelas anggota, petugas, pengurus, dan ketua kelompok*

Apa tugas dari masing – masing orang yang terlibat?

**Jawab :**

*Petugas tugasnya mengelolah unit simpan pinjam, pengurus itu lebih pada persetujuan kredit. Kalau ketua kelompok lebih sebagai perwakilan KPRI jadi yang mengkordinasi setoran bulanan misalnya*

Apa permasalahan yang ada dengan proses bisnis yang ada saat ini?

**Jawab :**

*Permasalahan lebih ke sisi anggota. Mengingat kondisi kecamatan Sumbermanjing Wetan yang berbukit dan luas sedangkan letak kantor KPRI yang berada di bagian paling utara kecamatan banyak anggota yang merasa berat untuk melakukan kativitas pada unit simpan pinjam. Untuk mendapatkan informasi simpanan atau pinjaman mereka harus mengeluarkan waktu dan biaya yang tidak sedikit. Apa lagi untuk pengajuan pinjaman yang paling tidak 3 kali datang ke kantor selama proses pengajuannya. Yang ditakutkan adalah anggota mulai berpindah ke koperasi – koperasi desa yang mulai bermunculan. Karena anggota adalah aset berharga bagi kami.*

Berapa anggota dari unit simpan pinjam saat ini?

**Jawab :**

*Untuk saat ini ada sekitar lebih dari 400 orang yang terbagi menjadi 70an kelompok atau instansi.*

Berapa banyak pengajuan yang biasanya ada pada tiap bulannya?

**Jawab :**

*Untuk jumlah pasti tidak dapat dipastikan, ada saatnya pengajuan banyak biasanya mendekati tahun ajaran baru. Paling tidak ada 3 atau 4 pengajuan dalam 1 bulan.*

Jika dibuat sebuah sistem informasi simpan pinjam, fitur apakah yang harus ada didalamnya?

**Jawab :**

*Mungkin untuk saat ini menu – menu utama yang harus ada seperti melihat simpanaan atau pinjaman sehingga anggota tidak kesusahan untuk mendapatkan informasi. Selain itu mungkin menu untuk melakukan pengajuan pinjaman sehingga mempermudah anggota. Ya mungkin seperti internet banking tapi lebih sederhana saja.*

Jika sistem informasi simpan pinjam ada, apakah anggota dapat mengakses sistem tersebut?

**Jawab :**

*Dilihat dari kondisi saat ini bisa, karena hampir seluruh guru yang merupakan anggota telah terbiasa menggunakan smartphone misalnya. Kebanyakan saat ini penyampaian informasi atau berita antar guru sudah menggunakan WA. Saya rasa saat ini anggota sudah banyak yang mengenal dan mengetahui cara mengakses sebuah web misalnya.*

Bagaimana kesiapan unit simpan pinjam untuk menggunakan sistem informasi baru?

**Jawab :**

*Saya rasa dari sisi petugas dan peralatan dari unit simpan pinjam sudah memadai. Petugas juga sudah menggunakan komputer selama ini tapi masih karena penyimpanan data kita menggunakan Excel dan Word. Ya jadi nanti hanya perlu sedikit pelatihan dan pengenalan untuk penggunaan sistem baru.*



Pengawas Unit Simpan Pinjam

Dr. Arifin Ihsan



## LAMPIRAN PENGUJIAN UAT

### User Acceptance Testing

SISTEM INFORMASI CUSTOMER - TOUCHING APPLICATIONS UNIT SIMPAN PINJAM KPRI  
"GURU" KECAMATAN SUMBERMANJING WETAN

Nama :

Instansi :

Jawab pertanyaan dibawah menggunakan nilai :

TS : Tidak Setuju S : Setuju

No	Pertanyaan	Jawaban	
		TS	S
1.	Sistem Informasi Simpan Pinjam KPRI dapat diakses dengan cepat.		
2.	Anggota dapat memperoleh informasi simpan pinjam KPRI pada sistem informasi dengan mudah.		
3.	Anggota merasa senang dengan tampilan Sistem Informasi simpan pinjam.		
4.	Adanya Sistem Informasi Simpan Pinjam mempermudah aktifitas dalam unit simpan pinjam.		
5.	Adanya fitur estimasi pinjaman mempermudah anggota untuk menentukan pengajuan pinjaman.		
6.	Sistem Informasi simpan pinjam meningkatkan pelayanan kepada anggota.		
7.	Adanya fitur pengisian form pinjaman online mempermudah anggota dalam melakukan pengajuan pinjaman.		
8.	Fitur estimasi mempunyai tingkat akurasi mendekati perhitungan sebenarnya.		

Malang, \_\_\_\_\_

## User Acceptance Testing

SISTEM INFORMASI CUSTOMER - TOUCHING APPLICATIONS UNIT SIMPAN PINJAM KPRI  
"GURU" KECAMATAN SUMBERMANJING WETAN

Nama : *Tatik Suniars*  
 Instansi : *SDN Harjo Kuncaran 3*

Jawab pertanyaan dibawah menggunakan nilai :

TS : Tidak Setuju S : Setuju

No	Pertanyaan	Jawaban:	
		TS	S
1.	Sistem Informasi Simpan Pinjam KPRI dapat diakses dengan cepat.		✓
2.	Anggota dapat memperoleh informasi simpan pinjam KPRI pada sistem informasi dengan mudah.		✓
3.	Anggota merasa senang dengan tampilan Sistem Informasi simpan pinjam.		✓
4.	Adanya Sistem Informasi Simpan Pinjam mempermudah aktifitas dalam unit simpan pinjam.		✓
5.	Adanya fitur estimasi pinjaman mempermudah anggota untuk menentukan pengajuan pinjaman.		✓
6.	Sistem Informasi simpan pinjam meningkatkan pelayanan kepada anggota.		✓
7.	Adanya fitur pengisian form pinjaman online mempermudah anggota dalam melakukan pengajuan pinjaman.	✓	
8.	Fitur estimasi mempunyai tingkat akurasi mendekati perhitungan sebenarnya.		✓

Malang, *19.02.2016*

*Tatik Suniars*  
*Tatik Suniars*

## User Acceptance Testing

SISTEM INFORMASI CUSTOMER - TOUCHING APPLICATIONS UNIT SIMPAN PINJAM KPRI  
"GURU" KECAMATAN SUMBERMANJING WETAN

Nama : Drs. Mukono  
Instansi : SDN Ringinsari 1

Jawab pertanyaan dibawah menggunakan nilai :

TS : Tidak Setuju S : Setuju

No	Pertanyaan	Jawaban	
		TS	S
1.	Sistem Informasi Simpan Pinjam KPRI dapat diakses dengan cepat.		✓
2.	Anggota dapat memperoleh informasi simpan pinjam KPRI pada sistem informasi dengan mudah.		✓
3.	Anggota merasa senang dengan tampilan Sistem Informasi simpan pinjam.		✓
4.	Adanya Sistem Informasi Simpan Pinjam mempermudah aktifitas dalam unit simpan pinjam.		✓
5.	Adanya fitur estimasi pinjaman mempermudah anggota untuk menentukan pengajuan pinjaman.	✓	
6.	Sistem Informasi simpan pinjam meningkatkan pelayanan kepada anggota.		✓
7.	Adanya fitur pengisian form pinjaman online mempermudah anggota dalam melakukan pengajuan pinjaman.		✓
8.	Fitur estimasi mempunyai tingkat akurasi mendekati perhitungan sebenarnya.	✓	

Matang, 12 - 02 - 2018

Drs. Mukono

## User Acceptance Testing

SISTEM INFORMASI CUSTOMER - TOUCHING APPLICATIONS UNIT SIMPAN PINJAM KPRI  
"GURU" KECAMATAN SUMBERMANJING WETAN

Nama : WIDYANINGSIH

Instansi : SDN. SEKAHBANYU 2.

Jawab pertanyaan dibawah menggunakan nilai :

TS : Tidak Setuju S : Setuju

No	Pertanyaan	Jawaban	
		TS	S
1.	Sistem Informasi Simpan Pinjam KPRI dapat diakses dengan cepat.		✓
2.	Anggota dapat memperoleh informasi simpan pinjam KPRI pada sistem informasi dengan mudah.	✓	
3.	Anggota merasa senang dengan tampilan Sistem Informasi simpan pinjam.	✓	
4.	Adanya Sistem Informasi Simpan Pinjam mempermudah aktifitas dalam unit simpan pinjam.	✓	
5.	Adanya fitur estimasi pinjaman mempermudah anggota untuk menentukan pengajuan pinjaman.		✓
6.	Sistem Informasi simpan pinjam meningkatkan pelayanan kepada anggota.		✓
7.	Adanya fitur pengisian form pinjaman online mempermudah anggota dalam melakukan pengajuan pinjaman.		✓
8.	Fitur estimasi mempunyai tingkat akurasi mendekati perhitungan sebenarnya.		✓

Malang, 17-02-2018

  
 WIDYANINGSIH

## DAFTAR PUSTAKA

- Agiledata, 2018. *Agile/Evolutionary Data Modeling: From Domain Modeling to Physical Modeling*. [online] Tersedia di: <  
<http://agiledata.org/essays/agileDataModeling.html> > [Diakses 12 November 2017].
- Arifin, D. 2005. "Rancangan Aplikasi e-CRM pada PT Essensilindo Perdana". *Soft Computing, Intelligent Systems and Information Technology*.
- Paryitno,E., Amylia,N. 2017. "Positive Impact of CRM Implementation to Improving The Services of Animal Polyclinics Customers". IEEE.
- Becker, J., Kugeler, M. & Rosemann, M., 2013. *Process Management: A Guide for the Design of Business Processes*. Berlin: Springer-Verlag.
- Bittner, K. & Spence, I., 2002. *Use Case Modeling*. Boston: Addison Wesley.
- Booch, G., Rumbaugh, J. & Jacobson, I., 1998. *The Unified Modeling Language User Guide*. Boston: Addison Wesley.
- Booch, G., Rumbaugh, J. & Jacobson, I., 2007. *Object-Oriented Analysis and Design with Applications*. Boston: Addison Wesley.
- Buttle, Francis., 2009. *Customer Relationship Management: Concepts and Technology*, 2nd ed., Butterworth-Heinemann, Burlington.
- IBM, D. T., 1998. *Rational Unified Process, Best Practices for Software*. Cupertino: Rational Software.
- Jalote,P. 2008. *A Concise Introduction to Software Engineering*. Dehlhi: Department of Computer Science and Engineering IIT.
- Kalakota,R. & Robinson,M., 2001. *Roadmap for Success. e-Business 2.0*. Boston: Addison Wesley.
- Kinasih,A. 2017. "Pengembangan Sistem Evaluasi Pembelajaran PAUD (Studi Kasus di PAUD Seruni 05 Kota Malang)". J-PTIIK.
- Kshirasagar,N.,Tripathy,P. 2008. *Software Testing And Quality Assurance*. Boston: Addison Wesley.
- Leffingwell, D. & Widrig, D., 2003. *Managing Software Requirements: A Use Case Approach*. 2nd penyunt. Boston: Addison Wesley.
- Lucidchart, 2012. *Business Process Modeling Software*. [online] Tersedia di : <  
<https://www.lucidchart.com/pages/business-process-modeling/software>> [Diakses 26 Oktober 2017].
- Object Management Group, 2011. *Business Process Model and Notation (BPMN)*. [pdf] Object Management Group.
- Paryitno,E., Amylia,N. 2017. "Positive Impact of CRM Implementation to Improving The Services of Animal Polyclinics Customers". IEEE.

- Payne, A., 2005. *Handbook of CRM*. Oxford: Butterworth-Heinemann.
- Pressman, R. S., 2010. *Software Engineering: A Practitioner's Approach*. 7th ed. New York: McGraw-Hill.
- Przybyłek, A., 2011. *Bridging the gap between business process models and use-case models*. Gdańsk, Gdańsk University of Technology.
- Reiner,K., 2010. *Introduction to Information Systems: Enabling and Transforming Business*. [e-book] John Wiley & Sons. Tersedia di: Google Books <<https://books.google.co.id/books?id=maK8xskEduYC>> [Diakses 21 Oktober 2017]
- Salehi,S. 2015. *Evaluation of the Effects of e-CRM on Customer Loyalty (Case Study: Esfahan Branch's of Sepah Bank)*. IEEE
- Sitio,A.& Tamba,H.,2001. *Koperasi: Teori dan Praktek*. Jakarta: Erlangga.
- Silver, B., 2012. *BPMN 2.0 Handbook*. 2<sup>nd</sup> ed. Florida: FutStrat.
- Sukamto, R.A. & Shalahuddin, M., 2014. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. 2nd ed. Bandung: Informatika.
- Sommerville, I., 2011. *Software Engineering*. 9th ed. Boston: Addison-Wesley.
- Taufik,M. 2015. "Analisis dan Implementasi Perancangan Metode Rational Unified Process pada layanan SDB dan Metode Pengujian Product Metric pada Bank Mandiri Cabang Palu Sam Ratulangi". *e-Proceeding of Engineering telkomuniversity.ac.id*.
- Zulfata,G.2017 "Pengembangan Electronic Customer Relationship Management (E-CRM) Pada Toko Sepatu Aneka Sport Malang Dengan Metode Kerangka Kerja Dynamic CRM ". J-PTIIK.